1988

An evaluation of automating Carrier Air Traffic Control Center (CATCC) status boards utilizing voice recognition input

Jensen, Robert D.; Spegele, John J.

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

AN EVALUATION OF AUTOMATING CARRIER AIR TRAFFIC CONTROL CENTER (CATCC) STATUS BOARDS UTILIZING VOICE RECOGNITION INPUT

by

Robert D. Jensen and John J. Spegele

June 1988

Thesis Co-Advisors: Dr. Gary K. Poock
Dr. Vincent Y. Lum

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a REPORT SECURITY CLASSIFICATION Unclassified	1b RESTRICTIVE MARKINGS
2a SECURITY CLASSIFICATION AUTHORITY	3 DISTRIBUTION / AVAILABILITY OF REPORT Approved for public release; Distribution is unlimited
2b DECLASSIFICATION / DOWNGRADING SCHEDULE	
4 PERFORMING ORGANIZATION REPORT NUMBER(S)	5 MONITORING ORGANIZATION REPORT NUMBER(S)
6a NAME OF PERFORMING ORGANIZATION Naval Postgraduate School 6b OFFICE SYMBOL (If applicable) Code 54	7a. NAME OF MONITORING ORGANIZATION Naval Postgraduate School
6c. ADDRESS (City, State, and ZIP Code) Monterey, California 93943-5000	7b. ADDRESS (City, State, and ZIP Code) Monterey, California 93943-5000
8a. NAME OF FUNDING / SPONSORING ORGANIZATION 8b OFFICE SYMBOL (If applicable)	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State, and ZIP Code)	10 SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO / PROJECT NO. / TASK NO / WORK UNIT ACCESSION NO

11 TITLE (Include Security Classification)
AN EVALUATION OF AUTOMATING CARRIER AIR TRAFFIC CONTROL CENTER (CATCC) STATUS BOARDS UTILIZING VOICE RECOGNITION INPUT

12 PERSONAL AUTHOR(S)
Jensen, Robert D. and Spegele, John J.

13a. TYPE OF REPORT	13b TIME COVERED	14 DATE OF REPORT (Year, Month, Day)	15 PAGE COUNT
Master's Thesis	FROM ______ TO ______	1988 June	177

16 SUPPLEMENTARY NOTATION
The views expressed in this thesis are those of the authors and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17 COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Connected-speech recognition; Voice recognition; Connected-speech syntaxes; Carrier Air Traffic Control (CATCC); Military speech recognition

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

Conducting safe flight operations from aircraft carriers requires accurate and timely dissemination of aircraft status information from the Carrier Air Traffic Control Center (CATCC). Presently, the information is manually displayed on status boards throughout the ship by a network of sailors communicating via sound-powered microphones. A prototype, connected speech-based system, developed by the Naval Ocean Systems Command (NOSC), was evaluated. Specific evaluation criteria were the hardware, software and the man-machine interface. The use of connected speech as an input modality across varying noise and syntactic conditions was experimentally tested. The result of this research was the proposal of guidelines for designing connected speech syntaxes and specific recommendations for future prototype development efforts.

20 DISTRIBUTION / AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION Unclassified
22a NAME OF RESPONSIBLE INDIVIDUAL Professor Gary Poock	22b TELEPHONE (Include Area Code) (408) 646-2636 / 22c OFFICE SYMBOL Code 55Pk

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

*U.S. Government Printing Office: 1986—606-24.

Approved for public release; distribution is unlimited.

An Evaluation of Automating Carrier Air Traffic Control Center (CATCC) Status Boards Utilizing Voice Recognition Input

by

John J. Spegele
Captain, United States Marine Corps
B.S., United States Naval Academy, 1978

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN INFORMATION SYSTEMS
and
MASTER OF SCIENCE IN COMPUTER SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL
June 1988

An Evaluation of Automating Carrier Air Traffic Control Center (CATCC) Status Boards Utilizing Voice Recognition Input

by

Robert D. Jensen
Captain, United States Marine Corps
B.A., Doane College, 1980

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN INFORMATION SYSTEMS

from the

NAVAL POSTGRADUATE SCHOOL
September 1988

James M. Fremgen, Acting Dean of
Information and Policy Sciences

ABSTRACT

Conducting safe flight operations from aircraft carriers requires accurate and timely dissemination of aircraft status information from the Carrier Air Traffic Control Center (CATCC). Presently, the information is manually displayed on status boards throughout the ship by a network of sailors communicating via sound-powered microphones. A prototype, connected, speech-based system, developed by the Naval Ocean Systems Command (NOSC), was evaluated. Specific evaluation criteria were the hardware, software, and the man-machine interface. The use of connected speech as an input modality across varying noise and syntactic conditions was experimentally tested. The result of this research was the proposal of guidelines for designing connected speech syntaxes and specific recommendations for future prototype development efforts.

TABLE OF CONTENTS

ACKNOWLEDGMENTS

We want to acknowledge the outstanding support provided by members of the Naval Ocean Systems Command, San Diego. In particular, the constant good humor and responsiveness to our numerous requests by Elaine Schiller and Bruce Plutchak were greatly appreciated. Additionally, the support provided by Dr. H. Koble and Jeff Nichols of ITT'S Defense Communication Division allowed us to remain on a very tight schedule and complete the evaluation as planned. Without their timely assistance, this effort would have been severely hampered.

I. INTRODUCTION

A. GENERAL

Managing, maintaining, interpreting, and displaying information is of critical importance to the safe and efficient operation of aircraft from a Naval aircraft carrier. The successful execution of the carrier's mission is largely dependent upon the ability to rapidly and safely launch, track, and recover high-performance aircraft operating from the carrier's deck. This thesis describes the research and evaluation of an automated information system designed to improve the present manual method of maintaining and displaying aircraft status information in direct support of aircraft launch and recovery operations.

B. PROJECT BACKGROUND

1. Purpose

The Naval Ocean System Command (NOSC), located in San Diego, California, developed a prototype information system to replace the current manual method of maintaining status board information in the Carrier Air Traffic Control Center (CATCC). The primary objective is to implement a system which will automate the maintenance, display, and distribution of aircraft status information using voice and/or keyboard as the input modality.

2. Key Participants

The primary participants in the project and their responsibilities were:

Activity	Responsibility
NOSC (Code 44)	System design and development
NPS (Code 55)	Prototype evaluation
NavAir	Functional management
USS Constellation	Primary test site
ITT, Defense Comm. Div.	Technical support, as requested

3. Status

A preliminary functional description has been developed, upon which the prototype system is based. Software development and initial testing was conducted at NOSC, San Diego, based on the preliminary design efforts conducted at that activity. Following initial development, field testing and evaluation was conducted at the Naval Postgraduate School (NPS) prior to full-scale shipboard testing.

C. SCOPE

In coordination with the thesis advisor, the research domain was limited to three primary areas of interest. First, evaluate the prototype system as delivered by NOSC, San Diego. The specific purpose is to objectively evaluate the system by gaining "hands on" experience in training, testing, and operation of functional system components. The second area is to make a general determination concerning the feasibility of automating the current system using some combination of voice and keyboard data entry to a computer-based system. Finally, based on evaluation and empirical testing, specific recommendations for future project efforts are provided.

D. METHODOLOGY

This research was conducted using the following approach:

1. Review voice recognition technology.
2. Study the CATCC operating environment.
3. Gain experience using the NOSC prototype.
4. Train a small user population on the NOSC system.
5. Conduct an experiment to evaluate the installed system.
6. Analyze the results.
7. Make specific recommendations based on experiences and test results.

E. LIMITATIONS

The primary research area is limited to evaluating the NOSC prototype, as delivered. Modifications by NPS were limited to those required to accomplish specific test objectives. The research is limited in several areas. First, the system was not, during the course of this research, tested in an at-sea environment. Second, the skill level of the test subjects, although familiar with CATCC operations, is not expected to be at the level of the sailors participating in these operations on a day-to-day basis. Third, the system developed by NOSC is designed to meet the generic CATCC requirements. Operational peculiarities of a specific CATCC were not considered. Finally, the researchers were unable to visit a CATCC during flight operations in the conduct of the study. CATCC-experienced officers were used instead to provide a rudimentary insight into essential details.

F. ORGANIZATION OF THE THESIS

The general organization of the thesis is by major topical components which are divided into distinct chapters. Depending upon the

experience of the reader, chapters may be omitted without loss of continuity. Each chapter will be preceded by a chapter executive summary providing the reader an opportunity to judge the contents prior to reading. Following this brief introduction, Chapter II presents a primer on voice recognition systems written for those unfamiliar with the technology. Chapter III discusses the mission, organization, and operational environment of a typical CATCC. The fourth chapter introduces the NOSC prototype system, as delivered to NPS. System Testing may be found in the fifth chapter. Finally, Chapter VI contains recommendations and conclusions.

II. VOICE RECOGNITION PRIMER

This chapter is a basic introduction to a variety of voice technologies and techniques. Specific topics discussed include how speech recognizers work, categories of speech recognition, typical applications, design criteria and a tutorial on the development of connected phrase syntaxes.

A. SPEECH RECOGNITION TECHNOLOGY

1. Speech Composition

Human speech is a complex, well-defined process of conveying information. The process starts with the brain, which sends signals to those muscles and organs used to make speech. The formation of speech sounds then occurs and the process ends with interpretation by the listener. This section will provide a basic foundation for understanding the way speech is formed, the composition of the speech signal, and the informational components of speech.

The physical process of communicating is achieved by the interaction of lips, tongue, and teeth. Five types of speech sounds articulated in English are: [Ref. 1:p. 13]

1. **Plosives** which are sounds created by stopping the passage of air. An example is the letter "t" in the word "top."

2. **Fricatives** are caused by forming a narrow passage through which air may pass. The diphthong "th" in the word "their" is an example.

3. **Laterals** are sounds formed when the tongue touches the roof of the mouth. An example is the "l" in "launch."

4. **Trills** are caused by the rapid vibration of one of the articulators (lips, tongue, etc.). The letter "r" is a trill sound in some languages.

5. **Vowels** are those sounds made when unobstructed air passes over the vocal cords.

Human speech, then, consists of strings of phonemes, which are the atomic units of sound. Most spoken languages require between 20 and 60 phonemes [Ref. 2:p. 128]. Table 2.1, adapted from Reference 2, p. 127, contains the phonemes typically associated with English. Analysis of the phonemes required for a word viewed in isolation is not sufficient because word sounds change depending upon the location within a string of words. A language's phonological rules govern the phonemes associated with a specific word depending upon the other sounds immediately preceding and following the word.

TABLE 2.1

ENGLISH PHONEMES

beat	bit	bait	bet	bat	Bob	but	batter	bought
boat	book	boot	about	roses	bird	down	buy	boy
you	wit	rent	let	met	net	sing	pet	ten
kit	bet	debt	get	hat	fat	thing	sat	shut
vat	that	zoo	azure	church	judge	which	battle	bottom
button								

Speech understanding is not based on word sounds alone. Understanding requires not only knowledge about *what* was said but

also *how* it was said. Hearing phonemes is the basis for what was said. Interpreting the stress, tempo, placement, and duration of pauses and intonation implies how it was spoken. This process is termed *prosodics*. An example would be understanding the implication of the following sentences:

"I can see a head." vs. "I can see ahead."

The sentences contain identical sounds, yet the prosodics of speech avoids the obvious ambiguity caused if pauses were not considered in the interpretation of what was said. Frequently though, prosodics alone is insufficient for understanding, as in the case of poor enunciation. Resolution of ambiguity may also involve an understanding of the context in which a phrase was spoken, which is termed *pragmatics*.

Human speech is also governed by a structure we know as grammar. The grammatical structure is represented by a syntax. English syntax, for example, requires a proper sentence to be composed of a noun and a verb phrase. The syntactic rules, in conjunction with prosodics, govern how an utterance may be correctly spoken. Linguistic theory suggests the more complex the syntactic constructs, the more powerful the language.

The human process, then, of semantic analysis of speech is reliant upon not only hearing the strings of phonemes but also using the prosodics, pragmatics, and syntax of the language in order to understand not only what was said but also what was meant. This abil-

ity allows us to uniquely process phrases such as "up in arms" and "over the hill."

Depending upon the application, speech systems may offer varying degrees of sophistication—from the simple phoneme interpreter (an isolated word recognizer) to a system capable of resolving prosodic and semantic ambiguity (a natural language processor).

2. Speech Analysis

Understanding how speech is analyzed by a machine is simplified by developing parallels between the more familiar human process and the unfamiliar machine process. Figure 2.1 diagrams the fundamental components of any speech analyzer. A *Knowledge Source* is the relative maturity of the system, human or machine. Just as children can be "programmed" to understand, so can a machine. The sophistication or robustness of a speech analyzer then is directly related to its ability to process the variety of speech information (phonological rules, prosodics, syntax, and pragmatics).

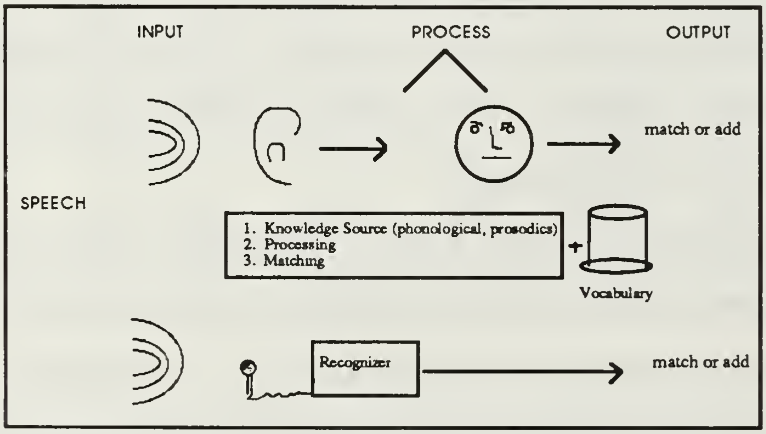


Figure 2.1

Speech Recognition Process

A fundamental algorithm for understanding what was said is found in Figure 2.2 [Ref. 3:p. 505]. This is a classic speech signal analysis algorithm that most processors use, regardless of the technology involved. Conversion of the human analog signal to a discrete digital signal in a machine-acceptable format is the first step. Once the signal has passed through the Analog-to-Digital converter, an attempt is made to bound the signal. Accurate detection of the boundaries of a signal is essential if recognition is to be achieved. Because the entire spectrum of the signal may not be required, an algorithm is employed to isolate the essential signal characteristics. The remainder of the signal is discarded in a process known as data compression. The probability that two utterances of a word or phrase are identical is remote. All recognizers, then, must be capable of eliminating slight variances in speech, pitch, intonation, and pause length. The filtering or "normalizing" process allows for a range of signal variability. The more robust the recognizer, the greater the variance. Depending upon the mode (learning or recognition), an attempt is made to either add the signal to a vocabulary or match the sound against an existing vocabulary.

Algorithms used to match the signal have been a major research area, with increasing both speed and accuracy a primary goal. Generally, though, matching is achieved by comparing distances between the incoming pattern and some previously stored reference pattern. The pattern with the minimum distance is judged the winner.

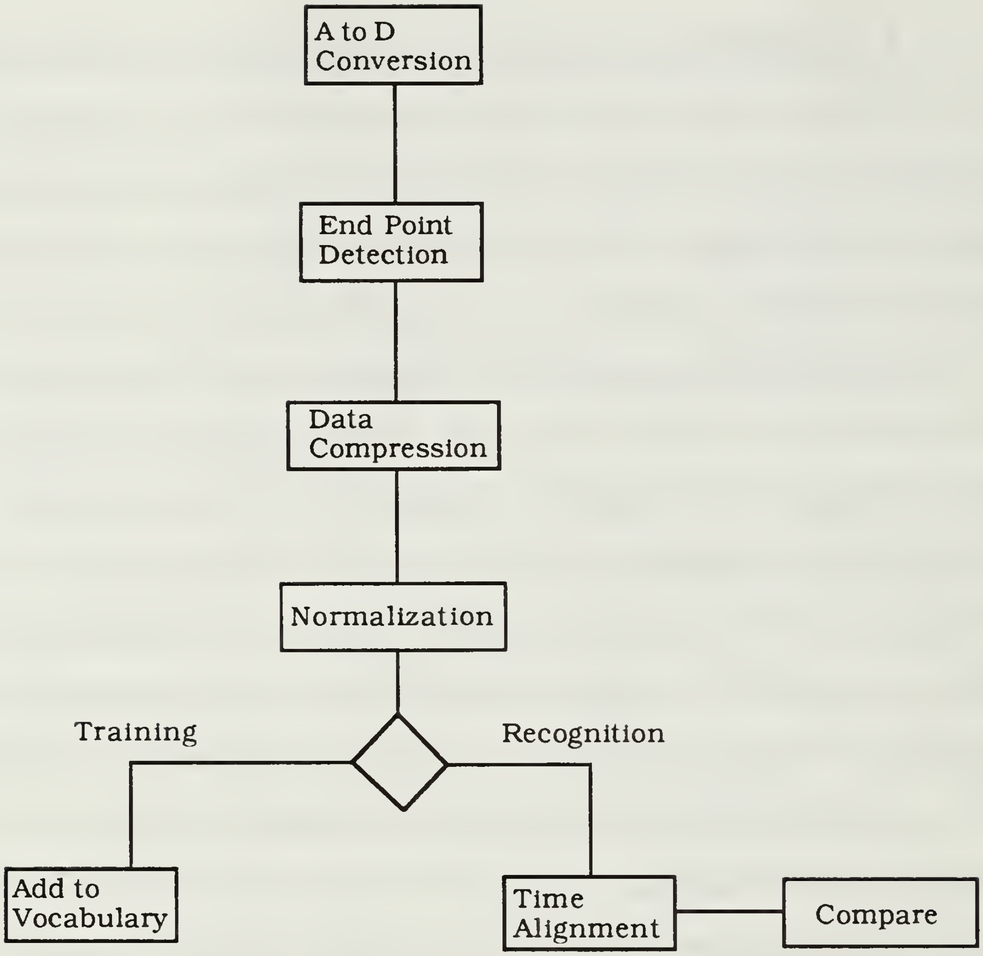


Figure 2.2

Speech Recognition Components

3. Categories of Recognizers

Research and commercial endeavors have combined to develop a variety of recognizers, which are designed to satisfy specific application requirements. Table 2.2 [Ref. 3:p. 503] compares and contrasts in simple terms the functionality of some of the most commonly found voice recognizer types. Two points to understand when evaluating any speech recognition system are the degree of speaker independence and how utterances are parsed.

TABLE 2.2

VOICE RECOGNITION CATEGORIES

Category	Speech Mode	Vocabulary Size	Language
Word Recognition (WR)	Isolated	$10 \rightarrow 300$	command-like
Connected Speech, Restricted (CSR)	Connected	$30 \rightarrow 500$	restricted command language
Speech Understanding (SU)	Connected	$100 \rightarrow 2,000$	English-like
Unrestricted Speech Understanding (USU)	Connected	$1,000 \rightarrow 10,000$	English-like
Unrestricted Speech	Connected	Unlimited	English

Speech systems today are either speaker independent or speaker dependent. The more common, speaker-dependent systems require the user to pre-train the system prior to use. Training typically involves creating a personal template signal for each word in the vocabulary. Creating a personal speech template for each word in the vocabulary ensures consistent input will be acceptable regardless of individual speaker characteristics. Unfortunately, for connected speech systems with large vocabularies, this could become a time-consuming process. Speaker-independent systems employ a standard template against which all speech is compared. The cost is generally a more restricted vocabulary and lower overall recognition rates.

Utterance parsing governs how the recognizer algorithm will dissect the utterance. In *isolated* systems the recognizer has no syntactic knowledge source, thus each utterance is viewed singularly. Examples would be the commands "ENTER" or "DIAL." Short macro

phrases are also possible in isolated systems. For example, a recognizer could be trained to recognize and subsequently execute the command "DIAL HOME." *Connected* systems, however, view the speech in terms of a syntax, thus strings of commands/words may be spoken in a connected pseudo-language that is a subset of a true language (e.g., English) for a particular environment (e.g., a CATCC). An example might be the command "DIAL PULSE FOUR ZERO EIGHT FIVE FIVE FIVE ONE TWO ONE TWO LOG IN GUEST." An extended variant of connected systems are those that recognize in a *continuous* fashion, typically according to some natural language syntax. Speech-to-text applications typically employ a continuous recognizer. As a general rule, the more powerful and complete the syntax is, the more natural the interface will be. Figure 2.3 summarizes the key differences between the competing approaches.

	ISOLATED	CONNECTED
SPEAKER-DEPENDENT	• simple to implement • low hardware cost • restricted to isolated utterances • high recognition rates	• increased training • short phrases to natural language • based on syntax
SPEAKER-INDEPENDENT	• limited application • small vocabulary • variable recognition rate	• most natural; powerful • response could be slow • recognition rates highly variable

Figure 2.3

<u>Speech Recognition Trade-Offs</u>

B. SPEECH APPLICATIONS

1. General

A parallel may be drawn between the development of telegraph/telephone systems and computers in general. When electronic communications was initially made possible, the input modality was via a key contact that transmitted a code representing a letter. Telegraph poles quickly out-paced the rival pony express and so this primitive keyboard became a means of communication within a system. The discovery by A. G. Bell that human speech could also be transmitted via wire caused the replacement of a keyboard with a voice-actuated receiver/transmitter as the primary means of transmitting short-duration messages.

Why did this occur? The primary reason is that despite our sophistication, voice remains our most natural communication medium. A prime example is how the US Navy has struggled with alternative mechanisms for over two centuries: signal flags with coded meanings, flares, and signal lights. But, given a choice, man generally prefers voice communication. Keyboards are an outgrowth of the typewriter and telegraph technologies but they, too, are limited by the skills of the operator.

Numerous studies have shown that voice recognition systems are faster and more accurate than most manual-entry systems. Additionally, voice systems free the operator's eyes and hands to accomplish concurrent tasks. Unencumbered by a keyboard or a mouse, the operator is generally free to move about while speaking to the system.

Table 2.3 [Ref. 4:p. 36] compares the relative advantages and disadvantages of speech in the military command and control environment.

In general, applications that are *most likely* to benefit from voice recognition input are those that have one or more of the following characteristics [Ref. 2:pp.4-8]:

1. Small working vocabulary
2. Well structured syntax
3. Operator's hands/eyes otherwise occupied
4. Reduced lighting conditions
5. Application requires other electronic communication (radio, telephone, etc.)

2. Commercial Applications

With the increasing sophistication and decreasing cost, commercial applications of speech systems have surfaced. The variety of applications is only limited by the imagination. But in the commercial environment, voice input is generally being used for one *primary purpose*: to increase individual productivity.

A typical commercial speech application is in the area of quality control and inspection. Such a system has been used by the Owens-Illinois Corporation since 1973. This isolated word application starts with the inspector entering, via voice, general shift information, employee number, and item type to be inspected. Then the operator conducts the inspection (hands occupied), calling out only the essential measurements. In a similar system, an automobile manufacturer

TABLE 2.3

ADVANTAGES AND DISADVANTAGES OF SPEECH I/O FOR C^2 ADVANTAGES

ADVANTAGES

Engineering

1. Can be faster than other modes of communication.
2. Can be more accurate than other modes of communication.
3. Compatible with existing communications systems.
4. Can reduce manpower requirements.

Psychological

1. Most natural form of human communications.
2. Best for group or team problem solving.
3. Universal (or nearly so) among humans.
4. Can reduce visual information overload.
5. Increase in value when also involved in cognitive-type processes.

Physiological

1. Requires less effort and gross motor activity than other modes.
2. Frees hands and eyes.
3. Permits multimodal operation.
4. Is feasible in reduced lighting.
5. Permits operator mobility.
6. Contains information about physical and emotional state of speaker.

DISADVANTAGES

Engineering

1. Interference from competing acoustic signals.
2. Environmental conditions can alter speech signal.
3. Requires use of microphone, a tool with which many users may not be familiar.

Psychological

1. Loss of privacy.
2. Psychologically induced changes in speech characteristics.

Physiological

1. Increased mental loading.
2. Fatigue from prolonged speaking.
3. Temporary physical ailments (e.g., colds, etc.) may alter speech characteristics.

drew the following conclusions about voice input following a two-week experiment [Ref. 5:p. 497]:

1. Voice recognition accuracy was at an acceptable level.
2. Minimal operator training (less than one day) was required.
3. Using the system did not interfere with task performance.
4. Operators were comfortable with the system.
5. A wireless microphone would allow complete operator freedom.

Other commercial applications that have been successfully installed include: voice applications of process control, warehousing functions, automated material handling, and parts programming for machine tools [Ref. 5:pp. 496–500]. Of particular importance are the environments in which these commercial systems have been used. Commercial applications have *not* been restricted to quiet, stable environments operated by a highly trained speech specialist. Rather, in many cases, these systems have been successfully introduced into such severe environments as airline baggage handling areas, assembly lines, factories, and warehouses.

3. <u>Military Applications</u>

The employment of speech in a number of mission-critical military systems has increased dramatically in the last decade. Speech recognition research and development has been largely supported by military organizations. Military speech recognition research efforts have been focused into three primary areas: command and control (C^2), messaging systems, and low-bit rate communications [Ref. 4:p.

35]. Because of the applicability to our study, we will focus our attention on military C^2 applications.

Increasing the sophistication of our combat systems has not come without a price. The multifunctional nature of the typical operational environment has dramatically increased the complexity of most systems fielded in the last decade. For example, today's high-performance tactical aircraft only remotely resemble their Korean- and even Vietnam-era counterparts. Aircrews are challenged by the increased complexity of the mission, which translates into an increased number of on-board systems requiring detailed attention. Each new system installed diverts the aircrew's attention from events outside the cockpit to those occurring inside. Aircrews today are nearly saturated with visual, aural, and manual input sources.

In the late 1970s, cockpit designers became aware of the problem and endeavored to improve the man-machine interface. Live test results illustrated advanced avionic systems for displaying information, heads-up displays (HUD), and the use of voice recognition. Sorely needed improvements to cockpit displays and systems combined with the HUD allowed members of the aircrew to focus their attention outside the cockpit. By using voice recognition, aircrews could query the status of specific mission-critical systems without having to reference cockpit displays. These test results, although not currently standard practice, showed that pilots using isolated word voice recognition commands could then aurally obtain airspeed, fuel-state, altitude, and ordnance information.

Again, these systems were employed in one of the most severe military environments. Designers have been able to overcome the combined effects of g-forces, vibration, and the distortion caused by oxygen masks, successfully implementing isolated word voice recognizers. Connected speech systems are the next generation to be installed, thus giving the aircrew an even more natural and flexible interface. Table 2.4 [Ref. 6:p. 310] delineates candidate military applications of speech technologies.

TABLE 2.4

POTENTIAL MILITARY APPLICATIONS OF VOICE I/O

1. SECURITY
   - Speaker verification
   - Speaker identification
   - Recognition of spoken codes
2. COMMAND AND CONTROL
   - System control (displays, fire control, aircraft)
   - Computer control
   - Material handling
   - Remote vehicle control
3. DATA TRANSMISSION AND COMMUNICATION
   - Speech synthesis
   - Scrambling/Ciphering
   - Messaging
4. PROCESSING DISTORTED SPEECH
   - Diver speech
   - Astronaut communication
   - Speech through protective or oxygen masks

C. DESIGN ISSUES

In section B, above, we highlighted some advantages and disadvantages of speech systems. With these in mind, we can start considering specific design issues. Figure 2.4 is a block diagram of specific issues identified by Lea [Ref. 2:p. 83]. We will examine each of Lea's issues in turn.

Figure 2.4

<u>Application Design Issues</u>

Application issues must be considered, as in the development of any system. These application criteria roughly equate to general design specifications. For example, what is the required response time? What is the minimum acceptable recognition rate? How reliable must the system be in terms of mean time between failures?

Human factors issues are of primary concern in most speech systems. If a clear advantage in terms of ease of use, accuracy, or efficiency can't be shown over alternative modalities, then perhaps voice input is not appropriate. The designer must consider human factors issues associated with training and the potential for problems in training users, particularly in a connected speech system, with a restricted syntax. Users, however, must be aware that the nominal time required to train a system is insignificant when compared to

long-term productivity growth. Regardless of how natural speech is, users may still resist a speech system, preferring instead a status quo alternative. Finally, and most importantly, any speech system must integrate the user into a well-developed system of displays with feedback available in both training and recognition modes.

Another design consideration of paramount importance is that of the language itself. For example, is there a well-structured vocabulary associated with the application? The application must also be studied in terms of the most appropriate class of recognition (isolated, connected, continuous). If a connected or continuous system is considered, the design will require development of an appropriate syntax.

Environmental conditions are of critical concern during the development of any speech application. The obvious concerns include noise [Ref. 7], vibration, lighting, and g-forces [Ref. 8]. Other concerns might be the impact of Electro-Magnetic Interference (EMI) on the channel itself.

Two performance-related issues are recognition accuracy and recognition tolerance. Recognition accuracy is a performance measurement expressed as a ratio of correctly spoken words/phrases to a base value. Recognition tolerance is the system's ability to correctly process speech under less-than-optimal conditions (e.g., stress, noise, g-forces, etc.). Additionally, Lea suggests the development and design of performance and evaluation tests. Will the test site accurately simulate expected operating conditions? How will the recognition be evaluated? What scoring methodology will be used? Finally, how will

voice input be measured and compared against alternative input systems and competing recognizers?

We will use the above issues as a foundation for our evaluation throughout the evaluation of the NOSC prototype system.

D. CONNECTED SPEECH GRAMMARS

The heart of any natural or near-natural language interface is the syntax of the system. In this section, we will formally introduce the concept and notation of a syntax and consider the development of connected speech grammars for two distinct classes of grammars: Natural Language (NL) Grammars and Phrase Grammars (PG).

The most powerful class of connected speech systems are those that accept natural language constructs as input. Natural Language Processing (NLP) has been a long-term goal of speech linguists. In the following subsection we will introduce the syntactic constructs necessary to support NLP.

A less powerful command-type application is the use of connected speech in the form of short phrases. NLP is at one extreme of the connected speech continuum. The less powerful syntaxes are designed to recognize short, command-type phrases. Structures for such systems we term phrase grammars (PG). Phrase grammars are tailored for each application, yet they are, in contrast to NL systems, much simpler to implement. The bulk of the research has been restricted to the NL systems; little research has been done in the area of design considerations for systems using PG. Following the NL grammar

section, we will propose specific evaluation criteria which may be applied to any PG-type system.

1. Syntax Terminology and Notation

A syntax, in simplistic terms, may be viewed as nothing more than a road map through a grammar. Borrowing from fundamental language theory, a syntax is represented by a set consisting of:

1. A start state
2. A set of final states (implying there may be multiple final states)
3. A set of intermediate states
4. Transitions between states

Figure 2.5 is a syntactic diagram for commands needed to play computer chess using speech input. The start state is the initial condition (usually silence). When an utterance is detected, an attempt is made to transition from the silence state (or node) to one of the follow-on states. The syntax, then, is the combination of legal utterances that lead from the start state to the final state. For example, in Figure 2.5 a legal utterance might be "MOVE ROOK TO QUEEN ROOK 3" or "STATUS CHECKMATE." The legality of the phrase is not guaranteed; it is, however, a syntactically correct utterance. The incorporation of intelligence in the syntax is a topic we will examine shortly.

2. Syntactic Analysis of Natural Language Grammars

Parsing the human language according to its grammatical constructs was the first technology that had to be developed before any NLP application could be fielded. Parsing is a technique by which the syntactic structure of an input may be analyzed. The primary

classes of parsers used to support syntactic analysis developed by linguists are: context free parsers, transformational parsers, and augmented context free parsers [Ref. 9:p. 22].

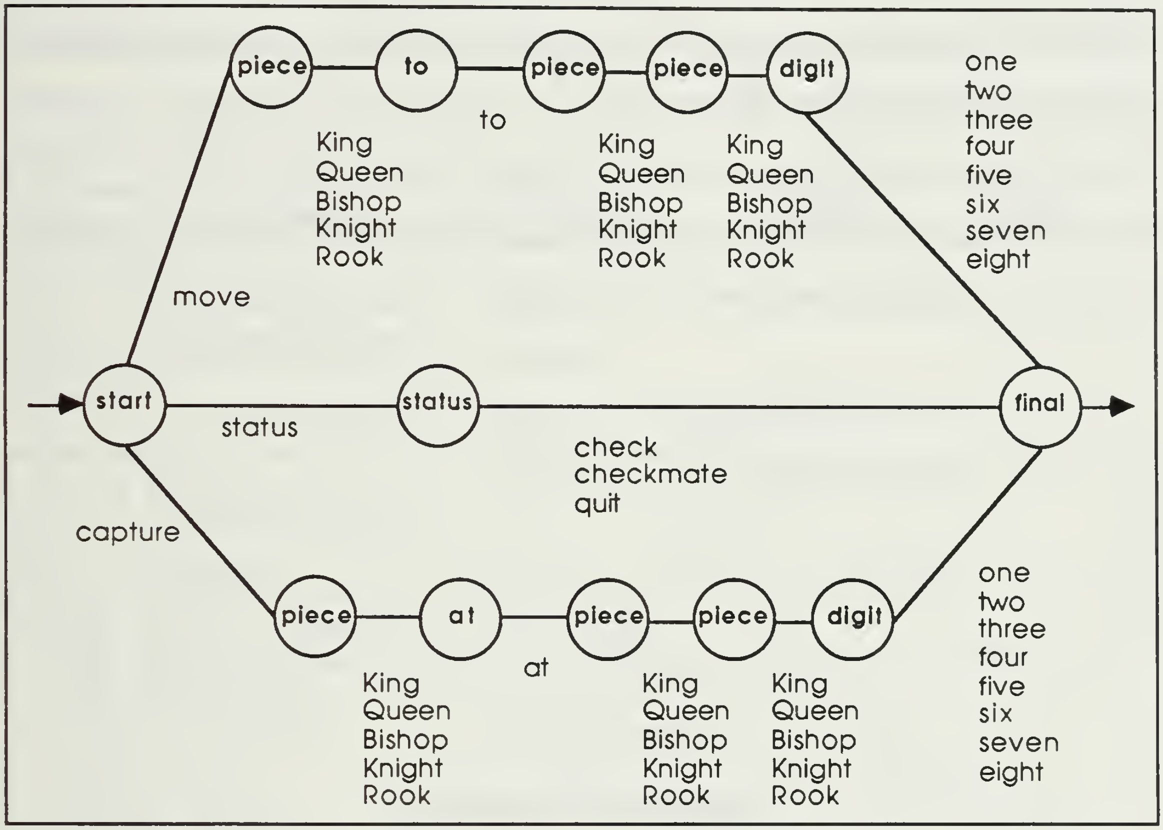


Figure 2.5

<u>**Sample Connected Speech Syntax**</u>

All language parsing techniques can be analyzed in terms of Chomsky's language hierarchy, first proposed in 1957. Figure 2.6 outlines the overall structure of the Chomsky hierarchy for representing grammars. Initial attempts at parsing human languages resulted in the development of phrase structured grammars which were identical to

Chomsky Regular Grammars. Linguists quickly discovered phrase structured grammars were a convenient means for representing a language that lacked the power to adequately describe a human language (English). We shall see later that although Regular Grammars weren't sufficiently powerful, linguists were able to modify the representation sufficiently to increase their power.

Type	Name	Format	Remarks
0	Unrestricted	$x \rightarrow y$	no restrictions most powerful
1	Context-sensitive	$x \rightarrow y$	where $\lvert y \rvert \geq \lvert x \rvert$
2	Context-free	$X \rightarrow y$	y is a terminal or a non-terminal
3	Regular	$X \rightarrow Yx$ $X \rightarrow x$	only productions allowed

Figure 2.6

Chomsky Heirarchy

Context Sensitive Grammars (CSGs), which are sufficiently powerful to represent NLs, were difficult to work with and were not used by language developers. A long-term argument developed over whether English required the power of a CSG, but this appears to have become a moot, theoretical discussion as developers have demonstrated reasonable success with alternative approaches to the problem of analyzing languages syntactically.

Context Free Grammars (CFGs) for many applications were the grammar of choice for developers attempting to model human language. Figure 2.7 [Ref. 10:pp. 225–232] is a much-simplified representation of an English language sentence structure with a representative syntactic decomposition of a simple English sentence. Artificial Intelligence languages such as Prolog are an effective mechanism for developing and analyzing the correctness of grammars developed. The conversion of a language from a CFG to Bacus-Nauer Form (BNF) and then to a Prolog format is relatively simple, as can be seen in Figure 2.8 [Ref. 9:pp. 73–79].

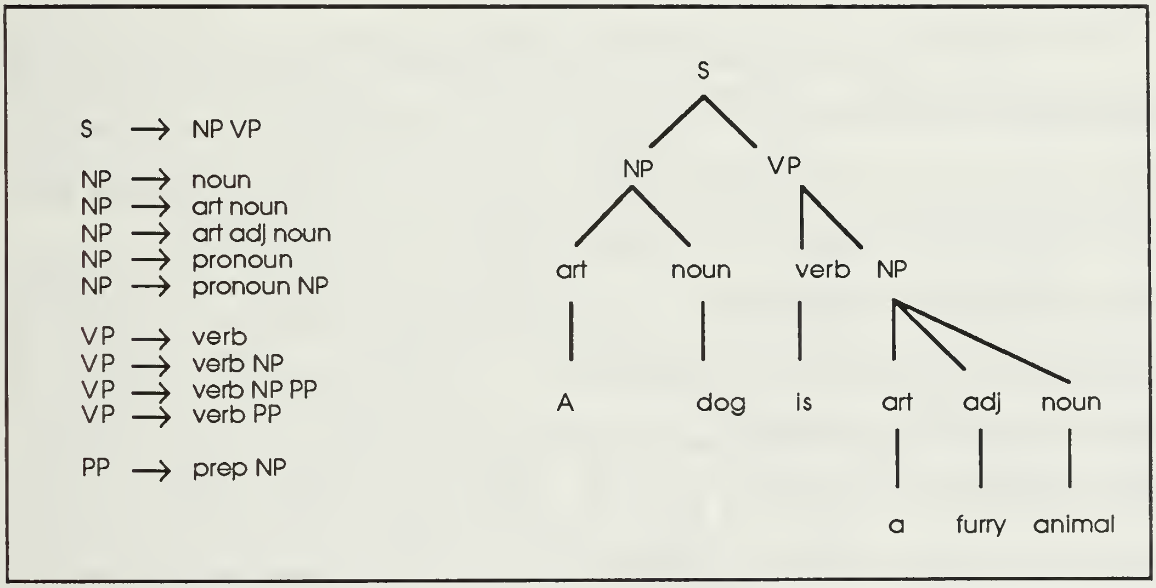


Figure 2.7

Simple CFG Grammar

BNF	CFG	PROLOG
<S> ::= <NP> <VP>	S → NP VP	s(X,Y):- np(X,Z),vp(Z,Y)
<NP> ::= noun | pronoun	NP → noun	np(X,Y):- noun(X,Y)
<NP> ::= art noun	NP → art noun	np(X,Y):- pronoun(X,Z),np(Z,Y)
<NP> ::= art adj* noun	NP → art adj noun	np(X,Y):- pronoun(X,Y)
<NP> ::= pronoun NP	NP → pronoun	np(X,Y):- art(X,W),adj(W,Z),noun(Z,Y)
	NP → pronoun NP	
<VP> ::= verb | verb PP |	VP → verb	vp(X,Y):- verb(X,Y)
verb NP	VP → verb PP	vp(X,Y):- verb(X,Z),np(Z,Y)
<VP> ::= verb | NP | PP	VP → verb NP PP	vp(X,Y):- verb(X,W),np(W,Z),pp(Z,Y)
	VP → verb PP	vp(X,Y):- verb(X,Z),pp(Z,Y)
<PP> ::= prep | NP	PP → prep NP	pp(X,Y):- prep(X,Z),np(Z,Y)

Figure 2.8

Alternative Representations of a Syntax

The next evolutionary step in grammar representation was transformational grammar. The notion of a transformational grammar, first proposed by Chomsky in 1957, grew out of a conviction that RGs and CFGs were insufficient to fully represent English (a concern that was later proved unfounded). A transformational grammar is based on a model consisting of two components: a base component, which is a CFG that generated additional or "deep structures"; and a transformational component, which is a set of rewrite rules. The primary problem with transformation grammars is that of combinatorial explosion [Ref. 11:pp. 151–162]. The parser must consider not a single path but rather a series of alternative paths which must be evaluated. Transformational grammars enjoyed only limited popularity and are rarely found in today's applications.

The long-term winner in parsing technology appears to be the approach with a basis in simple phrase structured grammars which are equivalent to RGs. Since regular grammars can be represented by finite state transition diagrams, linguists explored the possibility of expanding the power of these diagrams while retaining their simplicity.

The result is known as the transition network approach. A transition network is nothing more than a series of finite-state diagrams which are used to simulate the power of a CFG. The transition network consists of two components: a set of states and a set of arcs. Recursive transition networks (RTN) describe a language through recursion by developing a separate network for each non-terminal in the grammar. Figure 2.9, adapted from Allen [Ref. 12:pp. 41–46], is an RTN based on a simple subset of English grammar.

Developers were generally satisfied with the simplicity of the RTN but wanted to represent even more complex constructs. By adding the notion of registers to record the conditions and subsequent consequences of transiting an arc, they developed augmented recursive transition networks (ATNs). Kaplan claims that ATNs have the generative power of a Turing machine [Ref. 13:p. 83].

The apparent power and relative simplicity of ATNs has made them the overwhelming choice for developers for commercial NLU systems. Why is this so? The answer is directly related to the additional "status" information the ATN can maintain. Each network is

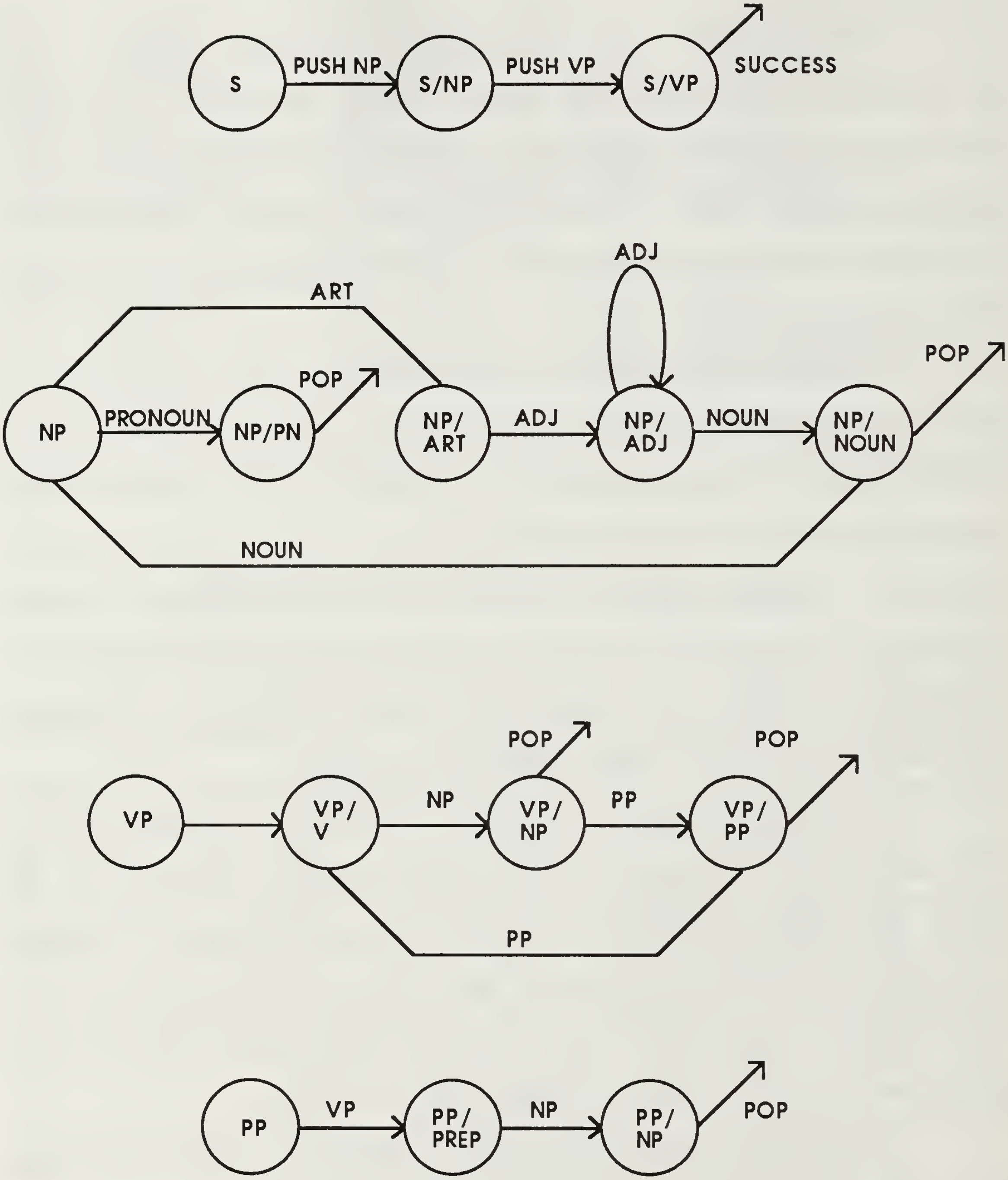


Figure 2.9

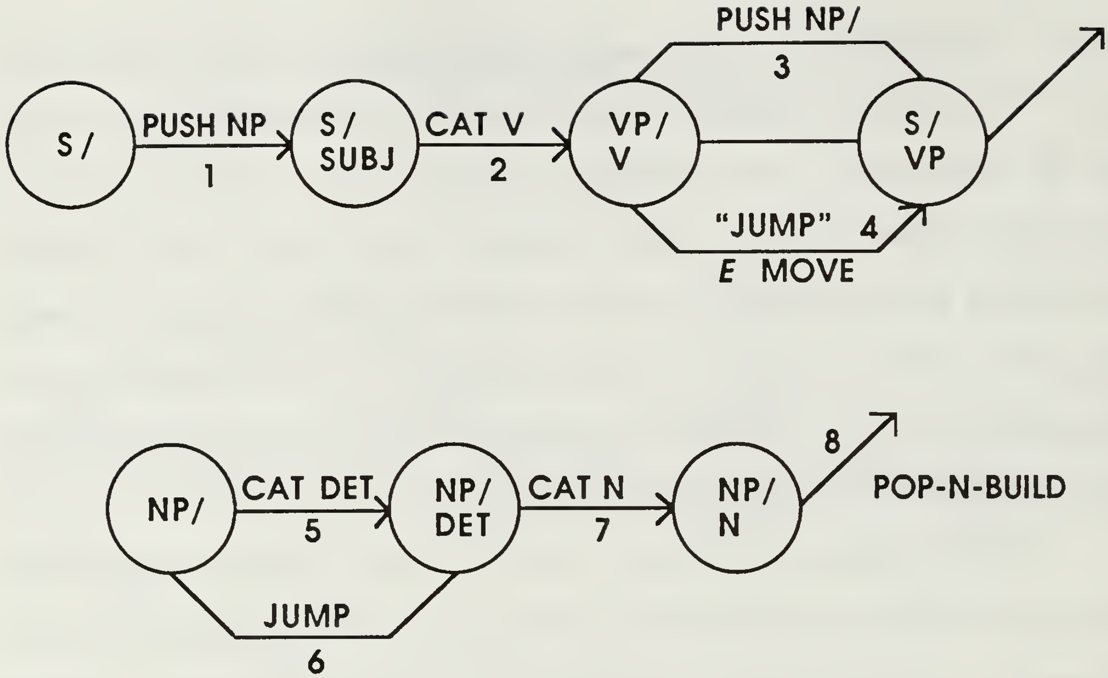
<u>**Recursive Transition Network**</u>

allowed to maintain a variety of registers while the local network is active. The registers maintain the status of specific syntactic conditions as they relate to the grammar being parsed. With this added power, the parser now is more intelligent and can, based on the grammatical rules and status of the registers, correctly parse the sentence. The best approach to understanding the functionality of an ATN parser is to trace through a sample sentence. Such an annotated sample is provided in Figure 2.10. [Ref. 13:p. 83]

While ATNs have shown to be the most promising approach to the NLP syntax problem, they, like any other RG, can only be programmed to accept valid, grammatically correct sentences. Poorly formed yet meaningful sentences cannot be supported. This inability to accept poorly formed or ambiguous input streams highlights the limitation of syntactic parsing of a sentence. Linguists discovered the parsing could only determine the *structure* of what was said, not the *meaning* of the input. Another process, semantic analysis, is needed if NLP systems are to become sophisticated enough to support the inherent ambiguities of a natural language.

3. Designing Phrase Syntaxes

Although less glamorous, the bulk of connected speech applications do not require the sophistication of NL grammars. Phrase-type grammars offer some advantages over a Natural Language system. First, PGs are simpler to implement. Second, restricting users to a small number of acceptable phrases eases the learning required.



TRANSITING ARC #	ACTIVE REG SET	INPUT REMAINING	REMARKS
1	NONE	The sailors drank grog.	Push current reg set. Enter NP.
5	DET Reg = The	sailors drank grog.	
7	CAT Reg = noun	drank grog.	Set person-num flag = plural.
8		grog.	Return to S/.
2	CAT Reg = verb	grog.	Set tense flag to past.
3		grog.	Push current.
6			Jump
7	CAT Reg = noun OBJ Reg = grog	–	

Figure 2.10

<u>**An Augmented Transition Network**</u>

Finally, inputs to a well-constructed PG with a limited vocabulary will be processed faster than a poorly designed NLP.

There are two ways to view speech system performance. The first is the traditional method of applying a scoring algorithm and assuming *all* errors are recognizer induced. The second approach is to assume the recognizer is capable of near-perfect recognition, viewing errors as syntax rather than recognizer failures. In analyzing the performance of a connected speech system, we must consider the isolated word and phrase scoring and resultant confusion matrix as a measure of the *system* performance, providing a window into the functioning of the syntax itself.

Despite the number of applications and the growing interest, the literature is silent on design considerations for developing PGs. In an attempt to fill the void, we have developed 10 rules for syntax development, summarized in Figure 2.11. These rules may be applied to either guide the design effort or analyze a syntax previously developed. Our objective in developing the rules was threefold. First, improve the recognition rate by avoiding syntax-induced errors. Second, improve processing performance by reducing the number of alternatives a recognizer must consider at each node. Third, incorporate human factors into the syntactic design.

We will use, as an example, a syntax which might be found in a typical grocer's butcher shop. The original syntax is shown in Figure 2.12. After introducing each rule, we will, if warranted, provide an example.

1. Eliminate non-determinism.
2. Avoid phonemic rhymes within a node.
3. Minimize nodal branching factor.
4. Discourage indiscriminate self-looping.
5. Provide escape from any node.
6. Eliminate silent jumps to finish.
7. Eliminate nonsensical transitions.
8. Limit phrase length (7 ± 2).
9. Avoid redundancy.
10. Limit syntax to specific (singular) task.

Figure 2.11

Syntax Design Rules

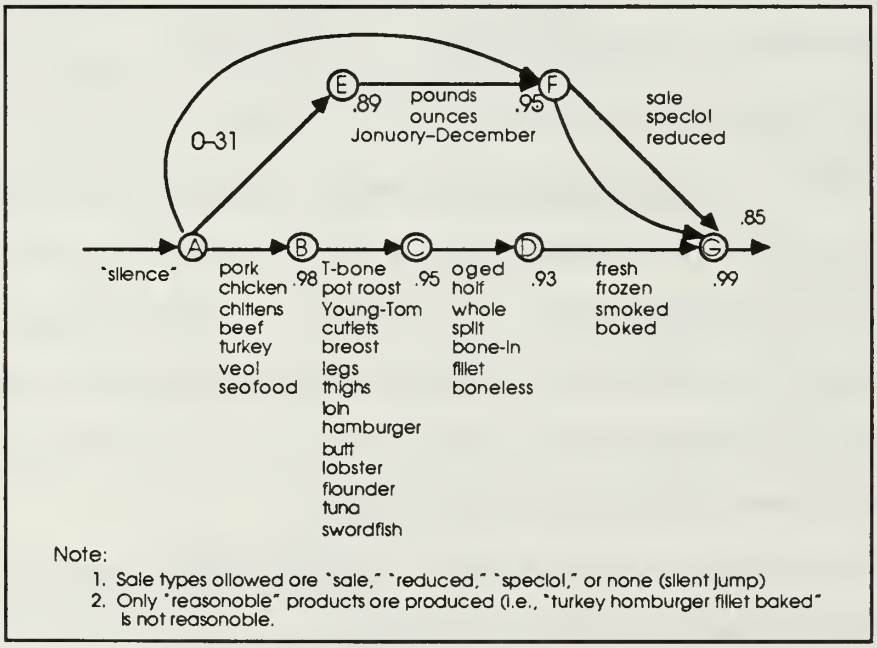


Figure 2.12

Butcher Shop Sample Syntax

a. **Eliminate Non-determinism**

Non-determinism occurs whenever an utterance appears on more than one path from a single node. Syntactic ambiguity is the result of non-determinism. A non-deterministic syntax cannot, by definition, be expected to perform correctly. Figure 2.13 is a partial syntax containing a non-deterministic ambiguity. Did the speaker intend orange to be a color or a fruit? Without knowing the context of the preceding and following utterances, it is impossible to interpret the intended meaning.

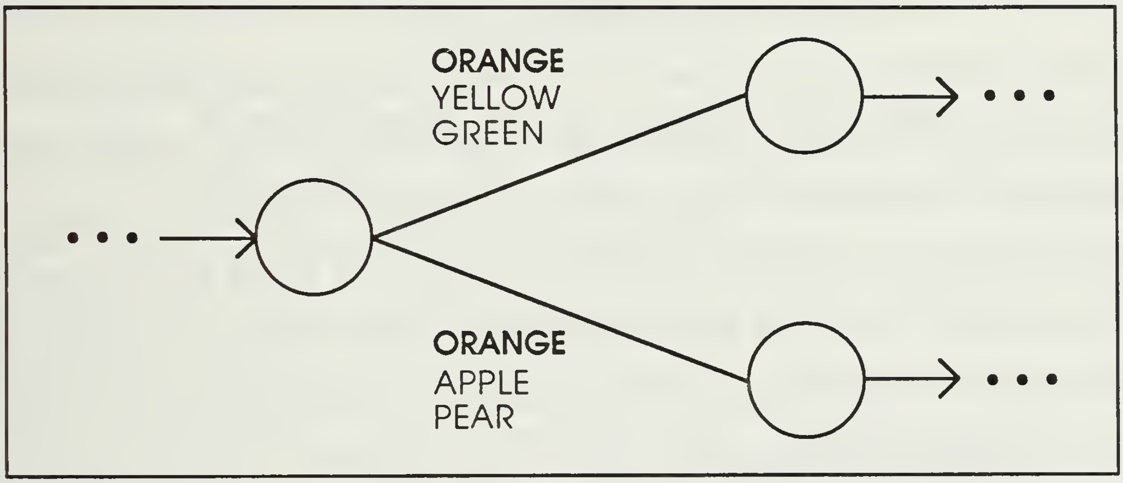


Figure 2.13

Non-Deterministic Ambiguity

b. **Avoid Phonemic Rhymes Within a Node**

Phonemic rhymes are a leading cause of substitution-type errors. Although sometimes unavoidable, words with similar phonemes should not be found in the same node. In our example, branching from the start state "CHICKEN" could easily be confused

with "CHITLENS" (depending upon speaker pronunciation). Eliminating the ambiguity can be achieved by finding a substitute word (POULTRY) or by reorganizing the syntax.

c. Minimize Nodal Branching Factor

As a general rule, the more word choices on a single branch, the greater the possibility for substitution errors. The smaller the branching factor, the better the performance. From node A in Figure 2.12, for example, we can transition on any of 39 utterances.

d. Discourage Indeterminate Self-Looping

A self-loop is used when multiple occurrences of the same set of utterances is desired. For example, the self-looping syntax in Figure 2.14 allows a single node to generate a string of digits with imbedded characters, ending with a character. Because the self-loop is indeterminate, the only known fact is that the string will consist of at least one digit and one character. Indiscriminate self-looping increases the branching factor, increasing the probability of an error. One approach to eliminating self-loops is to build separate nodes for the exact number of occurrences desired. Suppose, for example, an application required a phrase consisting of the last four digits of a social security number followed by the person's initials. While both syntaxes satisfy the specification, the bottom syntax in Figure 2.14 could be expected to have a higher probability of successful recognition. The exception to this is when self-loops are used as an error-correction technique or on a start node.

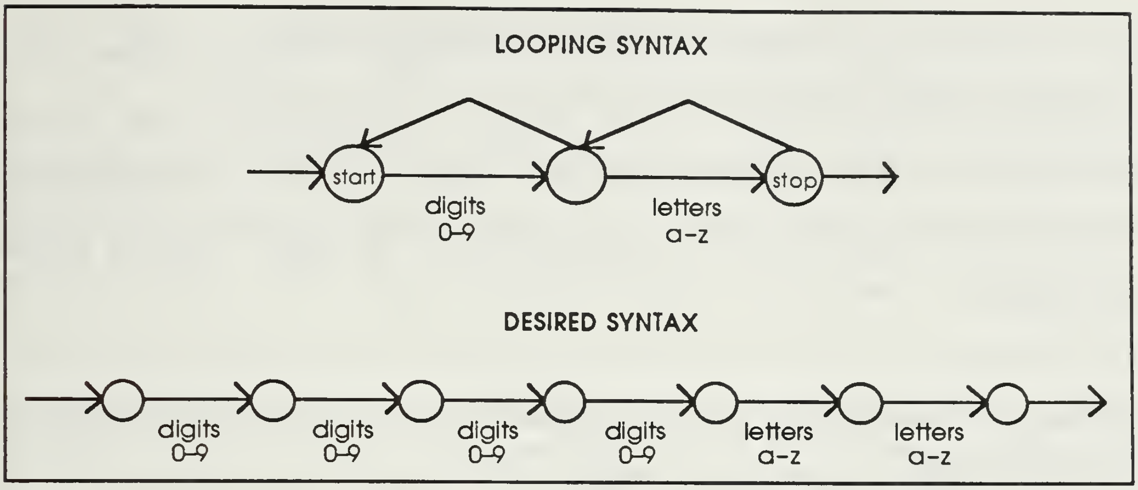


Figure 2.14

Self-Loop Removal

e. Provide Escape Mechanisms From Any Node

Frustration mounts when an utterance is misspoken (user error) or misrecognized (recognizer error), yet he/she is trapped by the syntax until they can "talk out" to the final state. Two correction techniques are to allow the user to either correct a single node immediately or to bail out and start over. Both techniques should be triggered by a single word (e.g., "CORRECTION" or "QUIT"). Figure 2.15 includes these escape mechanisms.

f. Eliminate "Silent" Jumps to the Final State

In a noisy environment, *any* noise may be potentially included as part of the input to the recognizer. Attempting to transition on "silence" to a state, particularly the final state, may result in substitution errors due to noise. Eliminate this problem by avoiding nodes which allow the user to follow a path through the syntax and then opt to transition on silence to the final state. If partial phrases

are acceptable, then they should be terminated with a unique utterance (e.g., "SEND," "OK," "STOP," etc.). In our example, the transition to the final state from the last node should be accomplished either with a adjective in the following node or some reserved terminator word.

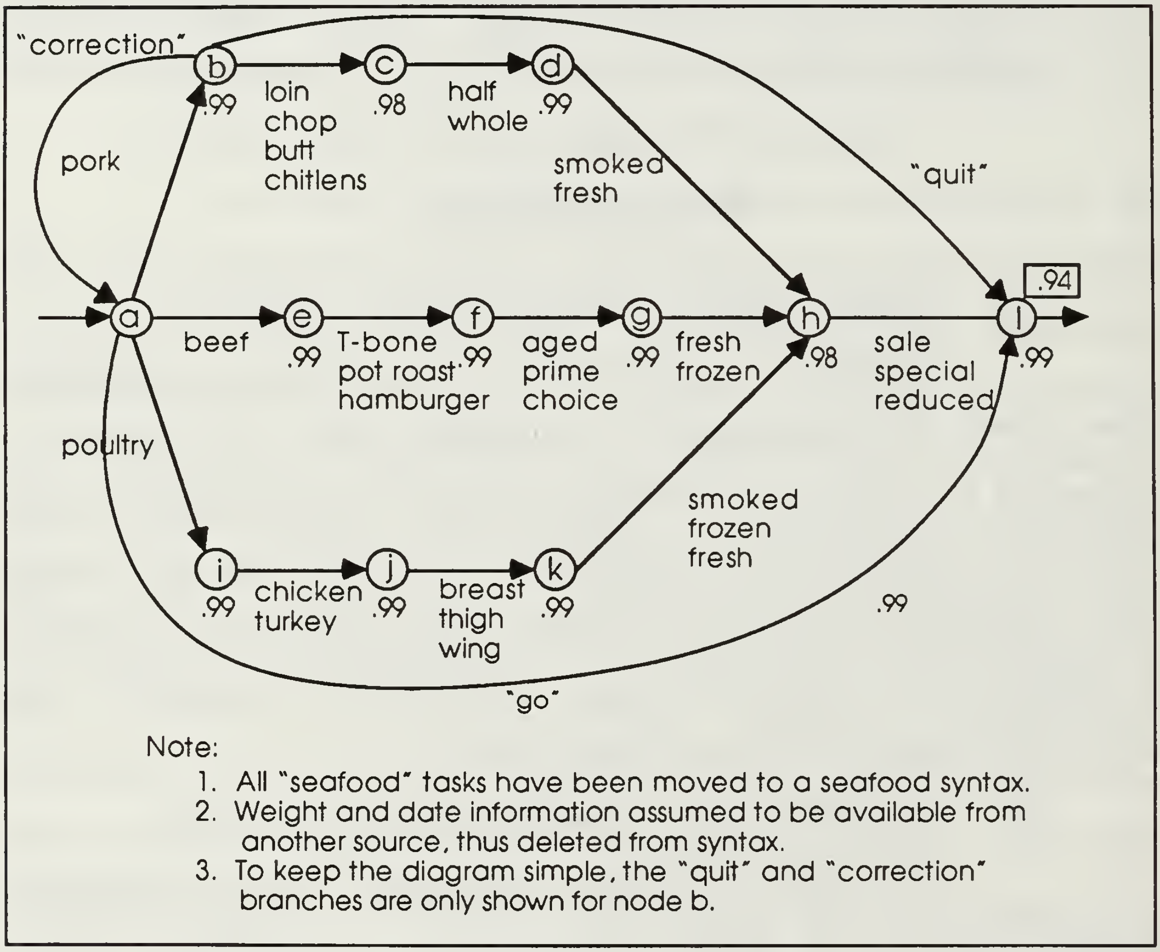


Figure 2.15

Corrected Syntax

g. Eliminate Nonsensical Transitions

In the course of developing the syntax, the designer is apt to improve flexibility by adding words in each node. The result is that certain combinations which do not occur in the "language" are still available in the syntax. These unused elements of a node will prove bothersome in the form of substitution errors. An example that comes to mind is in the use of digits. If there is a naturally occurring limit to a value in the application, then the digits node should be restricted to recognize only those digits which are possible. In our example, repeating the original digits node to represent ounces is nonsensical because a practical limitation to the ounces would be 15 (16 ounces being a pound).

h. Limit the Phrase Length to 7 ± 2 Words

Seven, plus or minus two, is a set of values frequently associated with human information processing capacity [Ref. 14:p. 52]. In this case, we propose it as a reasonable limit to phrase length. Two distinct problems are likely to occur with longer phrases. First, there is an increased probability of an error within the phrase. Remember that the probability of a recognition is an independent event; the total probability of speaking a phrase correctly is obtained by multiplying the probabilities of a correct recognition for each word by each other. Second, there can be an increase in operator-induced errors due to either incorrect syntax (phrase not allowed) or misspeaking. Lengthy phrases are unnatural and would logically be harder to learn; by enforcing a strict limitation on the phrase size we reduce the

probability the operator would become "tongue tied." One exception that frequently occurs is in the case of well-connected digits (e.g., a telephone number, social security number (SSN), etc.). Depending upon the content of the phrase, an entire string of digits might be considered a single word.

i. Avoid Redundancy

In order to increase system performance and to optimize the man-machine interface, voice input should not be used when alternative sources of information are available. For example, in our butcher shop there is little need for repeating the weight information displayed from an electronic scale. Ideally, the butcher would optimize the application by only using speech to identify the product and its attributes and use a scale to provide the weight information. Control would be obtained by capturing the weight at the command "GO." Generally, a phrase should only include information that is not available from other sources.

j. Limit Syntax to a Specific Task

Essentially, this rule suggests that the designer scope the syntax to a specific task. If there are related tasks with similar phrases, then we suggest a task-specific syntax be developed for each task. The fundamental concern is, again, with pruning the syntax so that only the essential, minimum set of transitions remains valid within the syntax. For example, in our butcher shop, suppose there were two separate lines maintained because of local sanitary restrictions—one for seafood only, the other for all other products. The

syntaxes for both applications, while similar, would have different vocabularies. The seafood line would not include references to beef or poultry products, with similar restrictions as appropriate for the "all others" line. It is important to note that while the vocabularies differ, the syntaxes should be as similar as possible since the same individual performs both tasks.

Figure 2.15 amplifies and supports the preceding discussion by redesigning the original syntax according to the ten design guidelines presented. Can we predict how dramatic the change would be? Suppose we assume that as the nodal branching factor increases by 3, the probability of a correct recognition decreases by 1 percent. We can then estimate the correct recognition probabilities for each node. Assuming we view the transition from each node as a discrete and independent event, we could estimate a comparative recognition probability for each syntactic phrase. These probabilities are found in the respective figures. The original syntax has a probability of approximately .84, while the redesigned syntax achieves an expected recognition rate of .94. The overriding concern in connected speech systems, then, is to strive for improved recognition rates through careful syntactic design.

III. CATCC OPERATIONS

This chapter is designed to introduce the reader unfamiliar with the Carrier Air Traffic Control Center (CATCC). Specific topics include mission description, organization, fundamental information flow, and the operating environment. The scope of this discussion will be limited to that background needed to understand the overall nature of the application. It is not intended to serve as a requirements definition or a functional description. Readers familiar with CATCC operations may omit this chapter without loss of continuity.

A. ORGANIZATION AND MISSION

The two primary organizations within the CATCC are: Air Operations (AirOps) and Carrier Controlled Approach (CCA). We will briefly examine both of these organizations.

The principal function of AirOps is to coordinate all flight operations for all airborne aircraft. Major tasks include:

1. Prepare the air plan.
2. Brief ready rooms.
3. Coordinate with divert airfields.
4. Monitor launch and recovery operations.
5. Maintain/display aircraft status and mission information, as required.
6. Coordinate diversion of airborne aircraft.

Figure 3.1 is a typical layout of AirOps. Pages 117 through 120 of Appendix A are sample layouts of the status boards and a description of the acronyms associated with each of the boards. AirOps is headed by an Air Operations Officer and manned with approximately eight sailors. Information needed to update status boards, internal and external to AirOps, is accomplished via operators using sound-powered communication systems. This information is duplicated for at least 50 individuals throughout the ship [Ref. 15:p. 1]. Frequent human-error and untimely transmission of the information throughout the ship adversely affects accomplishment of the AirOps mission.

The CCA's function is to provide for the safe and effective control of airborne aircraft. The CCA is specifically tasked with controlling all aircraft within a 50-mile radius of the carrier and for the recovery (i.e., safe landing on the carrier) of all aircraft operating under night and or Instrument Flight Rules (IFR) conditions. Major tasks include:

1. Control aircraft departures, marshal, approach and final approach.
2. Display and disseminate aircraft status information, as required.
3. Monitor launch and recovery operations.

CCA manning includes a CCA officer, assisted by a CCA supervisor. Additionally there are Marshal, Approach, Departure, and Final controllers. Approximately 10 individuals are needed to man the CCA. Information needed by other organizations is distributed via the same sound-powered phone system. Specific problems typically include

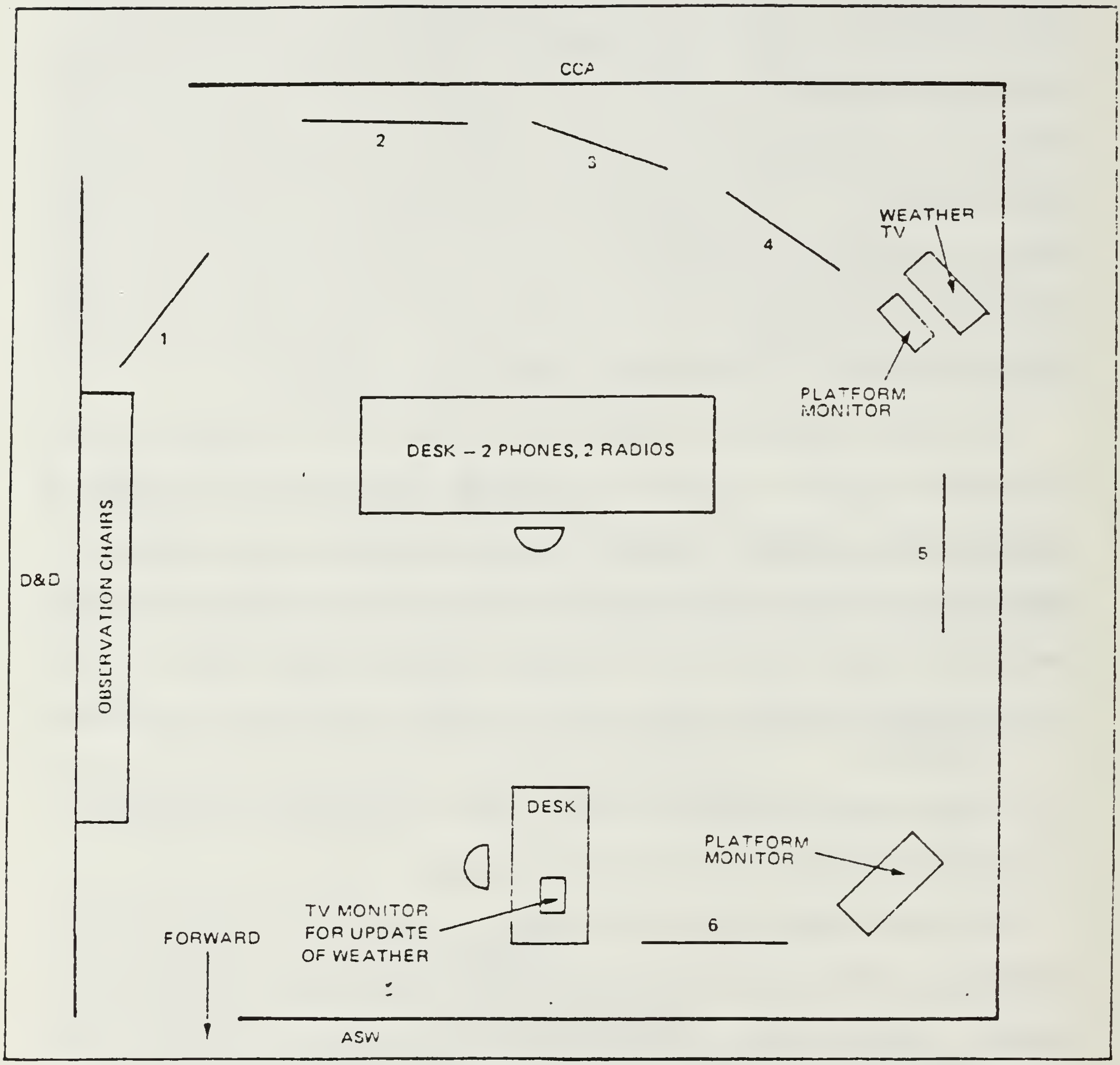


Figure 3.1

__Air Operations__

transferring fuel state and approach information to AirOps in a timely and accurate manner. Figure 3.2 is a layout of a typical CCA. Pages 121 through 127 of Appendix A are the status boards maintained by CCA personnel.

Marshal controller duties include tasks that ensure the orderly control and separation of aircraft awaiting approach to the carrier. The Marshal controller must issue to the approaching aircraft the following information:

1. Recovery type.
2. Marshal radial, distance, and altitude.
3. Expected Approach Time (ETA).
4. Time check.
5. Weather information.
6. Expected final bearing.
7. Approach frequency (button).

This information is also displayed in the CCA and communicated to other locations.

Departure and Approach controllers are responsible for the safe control of aircraft departing or approaching the ship. Information associated with these events includes departure or first-approach times, radio frequencies, aircraft status, and fuel state.

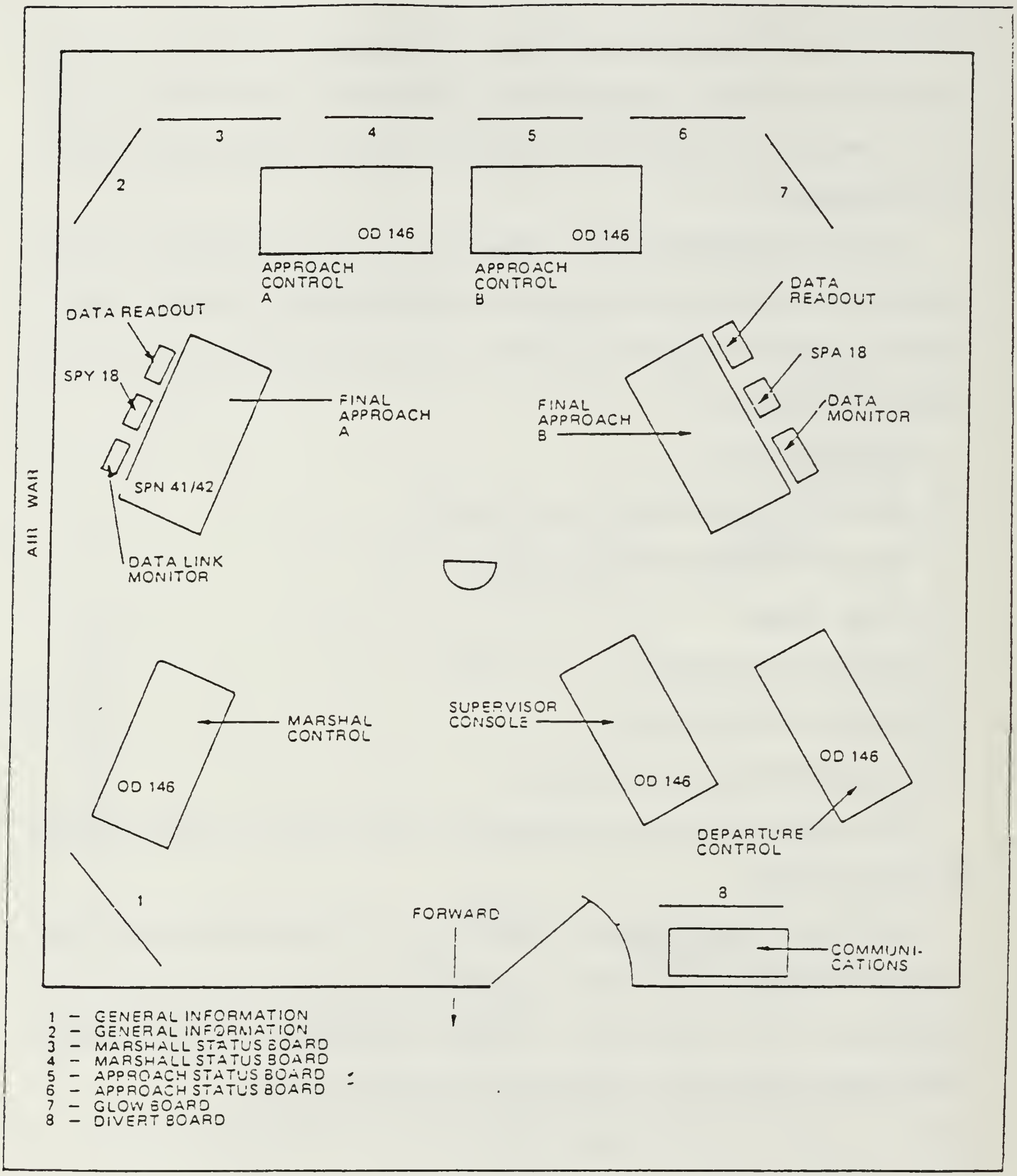


Figure 3.2

<u>Carrier Controlled Approach</u>

B. ENVIRONMENT

The at-sea operating environment is hostile to sensitive computer-based systems. In the following paragraphs we will examine some potential problems that can be anticipated in operating any system in the CATCC.

1. Power

Periodic fluctuations and losses of power are not an uncommon occurrence aboard any naval vessel. Commercial computer equipment, sensitive to power fluctuations, must have hardware and software protection systems to support unexpected losses or changes in current.

2. Vibration

A carrier during flight operations is subject to two kinds of periodic vibrations: (1) vibrations associated with being underway, and (2) vibrations caused by the launch and recovery of high-performance jet aircraft. Vibrations transmitted through deck plates and bulkheads affect all shipboard systems. Sensitive systems must be protected from vibration by a combination of ruggedization and shock mounting.

3. Illumination

The CATCC operates in a reduced lighting mode to enhance the contrast of radar displays. Operators must be able to operate their systems without the need for additional lighting.

4. Space

Space aboard a combatant vessel is at a premium. The CATCC is no exception. Discretionary space in the CATCC is at an absolute

minimum; there is barely sufficient area for the personnel and systems installed.

5. Noise

Noise sources in an operating CATCC include: Electronic "white" noise, noises associated with flight operations, radio transmission and other speaker noises, and human conversation.

6. Electro-Magnetic Interference (EMI)

The large number of electronic systems operating in close proximity are subject to spurious and unwanted EMI. The results of EMI, if not anticipated, are the unusual and seemingly inexplicable losses of data or changes in system operating characteristics.

7. Ventilation

Poor ventilation systems hinder the removal of heat generated by electronic components. Additionally, the lack of circulation hampers removal of smoke and dust particles which adversely affect sensitive devices such as magnetic tapes, magnetic diskettes, and computer read/write heads associated with mass storage devices.

The seven environmental factors alone are not sufficient when considering installation of a system at sea. Systems installed must, for instance, be able to withstand reasonable operator abuses (liquid spills, rough handling, etc.). Systems must also be capable of being maintained and operated by carrier personnel. Low-level maintenance of hardware and software should be able to be accomplished by embarked sailors as it is required. More extensive maintenance

requirements may require on-site contractor support at naval bases and repair facilities.

IV. THE NOSC PILOT SYSTEM

A. GENERAL

The pilot system designed and developed by NOSC (code 441) was intended primarily as a vehicle for validating the concept of automating updates to the CATCC status boards. The pilot system, as delivered, was not developed as the final solution to the application. It is, however, a first attempt at evaluating alternative architectures, application software, and voice recognition systems. From the pilot system, valuable insight useful for future prototype development, if warranted, can be obtained. It must be stressed that the system evaluated at NPS has not been installed in an operational at-sea test environment.

The hardware and software provided during the test may never be actually implemented in the final system. As key components in the pilot system, they are, nonetheless, valuable for establishing a baseline of experience upon which the application can be developed. In particular, we recognize that the installed voice recognition system and supporting software is a pre-production version made available to select research organizations. It is with that understanding that we examine the system, as installed, in the following sections. Except where noted, the system was intentionally evaluated "as delivered." Deviations were limited to those that would directly support the research effort.

The chapter will start with an overview of the hardware components, followed by a more detailed review of the recognizer used. We will then examine the software components and syntax design. This chapter will close with an orientation to the operational procedures involved in training and operating the system.

B. HARDWARE DESCRIPTION

1. Overview

The system is based upon a Sun Microsystems model 3/160 multi-user mini-computer. Configured around the successful VME architecture and supported by a Motorola 32-bit M68020 microprocessor, the Sun system is designed for, and capable of supporting, multiple users in a wide variety of applications. As delivered, the Sun system has a mass-storage device capable of storing 142 MB (megabytes) of information. In addition, the installed system was equipped with 8 MB of Random Access Memory (RAM). A mass-storage tape back-up was available for archiving files.

Connected to the system via RS-232 connectors were six WYSE model 60 ASCII terminals, three of which were equipped with standard keyboards for input. These terminals have a 14-inch amber-on-black display screen. The purpose of the each display will be covered when we discuss the status boards.

In addition to the six ASCII terminals, a Sun workstation was included. The Sun workstation is a black-on-white 19-inch display capable of supporting high-resolution graphics and multiple windows. This workstation is the primary terminal and was used in this

application for training, testing, and operation of the status boards. Associated with the terminal, was a light-driven mouse pointing device supported by the SUNTOOLS software. Menu selection and window control functions were the primary mouse-driven events.

Printed output was produced by a Texas Instruments dot-matrix printer connected to one of the Sun's printer ports.

NOSC provided Shure model 10 headsets with a Hewlett-Packard model 465A amplifier as recognizer voice input devices. These headsets, while suitable for low-noise conditions, proved unusable above 65 dBA of noise. A substitute Plantronics SNC 1436 noise-cancelling microphone was provided by ITT's Defense Communication Division (DCD) for the duration of the NPS evaluation. The use of the Plantronics headset eliminated the need for additional amplification of the input signal, allowing removal of the HP amplifier.

Figure 4.1 is an overall diagram of the hardware architecture we evaluated. We stress that this is only the initial configuration. The system may be expanded to include additional Sun computers, workstations, and display terminals supported via an Ethernet network. Figure 4.2, provided by NOSC, is a system architecture to which the system may ultimately evolve.

2. The Voice Recognition System

An ITT VRS 1280/VME was the voice recognizer included in the system. The VRS 1280 architecture includes its own M68000 processor and thus is not reliant on any external processor to support

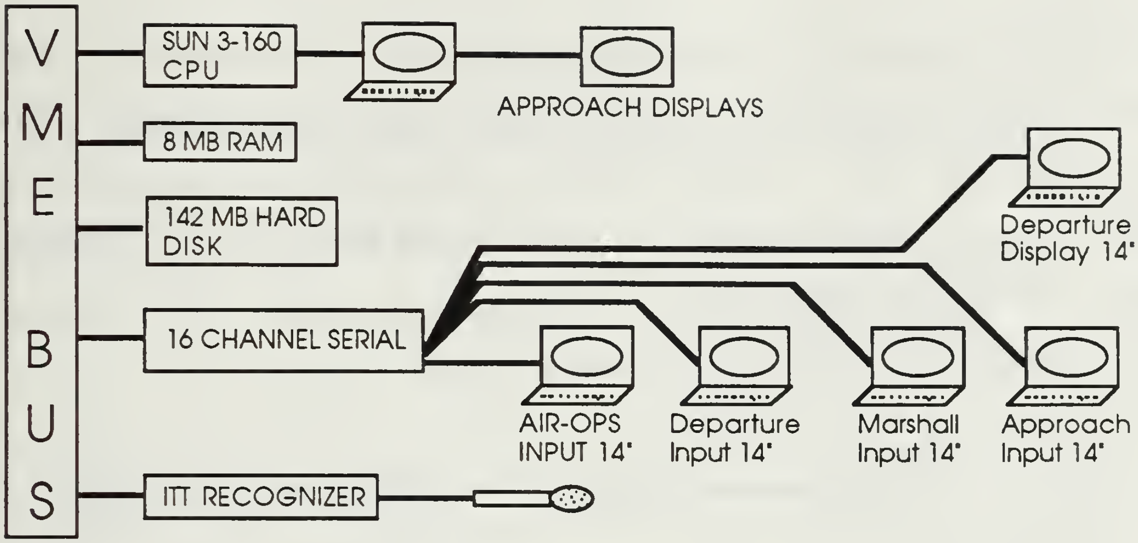


Figure 4.1

<u>**Pilot System Architecture, Tested**</u>

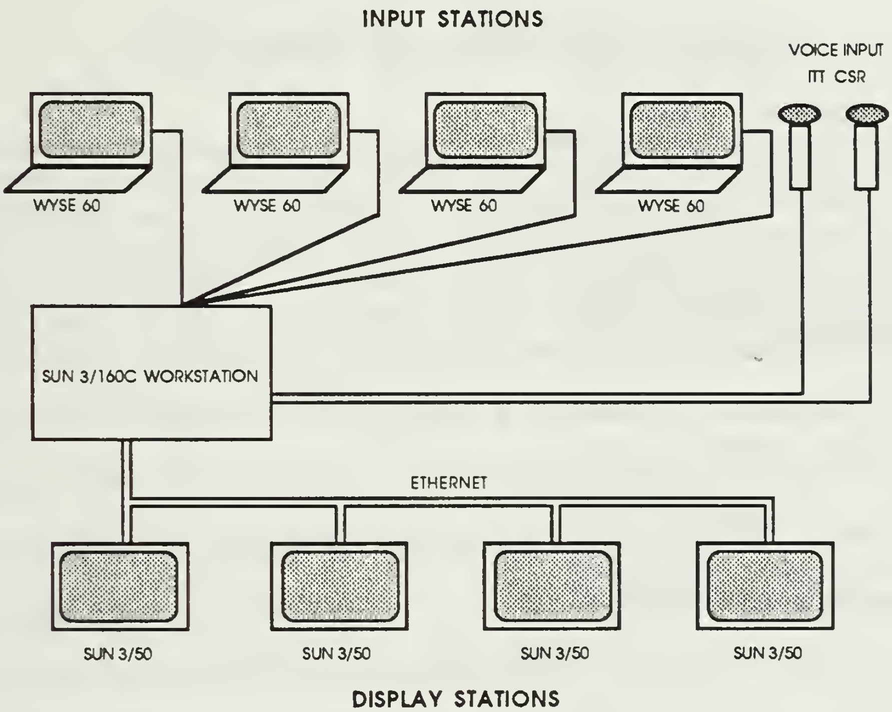


Figure 4.2

<u>**Electronic Status Board System Design**</u>

recognizer operations. The overall architecture is diagrammed in Figure 4.3 [Ref. 16:p. 11]. A summary of system features is found in Table 4.1 [Ref. 16:p. 14]. Template matching calculations are performed in the Dynamic Time Warping circuitry. While the exact technologies used by the recognizer are proprietary, the VRS 1280 Product Description does provide the following insight:

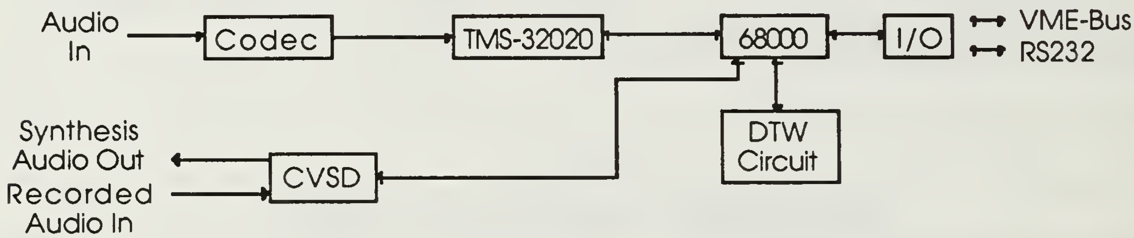


Figure 4.3

ITT DCD VRS 1280/VME Architecture

> ITTDCD's approach to speech and speaker recognition is based on a powerful kernel technology for the basic pattern matching algorithm. This kernel technology is referred to as the Template Determined Endpoint Detection (TDEP) algorithm....the ITT DCD algorithm does not employ any technique to explicitly detect where words begin and end prior to any pattern matching computations, thus eliminating a major source of recognition errors. [Ref. 16:p. 1]

A continuous matching algorithm compares the incoming signal against known vocabulary words, background noise templates, and phoneme templates, allowing for the identification of both speech and non-speech signals [Ref. 16:p. 2]. The syntax can be adjusted to support a variety of speech styles ranging from phrases without pauses to phrases with imbedded pauses of user-determinable length and location.

TABLE 4.1

MODEL VRS 1280/VME DETAILED SPECIFICATIONS

Recognition	Vocabulary Capacity:	• 500 unique words • 1280 sec. of speech (RAM) (approximately 2000 words)
	Throughput Capacity:	• >500 seconds of speech (approximately 800 words)
	Mode:	• Speaker dependent • Continuous or isolated words • Syntaxed as required
	Response Time:	• .25 second (avg.)
Training		• One or more repetitions of each vocabulary word for initial training • Easily updated if necessary to accommodate changes in the speaker's voice
Synthesis	Algorithm:	• CVSD
	Rate:	• 16 Kbps
	Capacity:	• 64 seconds of speech capacity in on-board RAM (additional vocabulary can be stored off-board) • Simultaneous with recognition
Record/Playback		• Record/playback function supported with CVSD analysis/synthesis; two 2-second buffers provided (also used for inputting messages to be synthesized • Simultaneous with recognition
Analog	In:	• Line input (0dbm, 600Ω)
	Out:	• Line output (0dbm, 600Ω)
I/O		• VME bus RS232
Physical	Size:	• Double-sized extended (233.3mm x 220mm) VME board form factor

C. SOFTWARE DESCRIPTION

Software for the system consists of both systems and applications software. The SUN computers operate under the UNIX operating system. In addition, the VRS 1280 is supported by ITT-supplied User Interface Software (UIS), which is a menu-driven system for interacting with the recognizer. Application programs, developed in the "C" programming language by NOSC, parse outputs from the recognizer and control the status board displays. In addition, a series of routines were developed to automate the menu selection process for training, testing, and operation of the ITT UIS.

Detailed discussion of the UNIX operating system is not required within the scope of this research. Specific NOSC applications, programs, and user routines will be discussed in detail in later sections. Important to the research, however, is an understanding of the functions available via the ITT UIS. Many of these functions were hidden from the user by NOSC routines developed to improve and simplify the user interface. Nonetheless, a rudimentary understanding of the ITT-supplied interface is considered necessary to understanding the functionality of the recognizer.

The UIS consists of user-selectable two-character commands presented in a series of menus. We will limit our discussion to the most important commands found in the main menu (Figure 4.4).

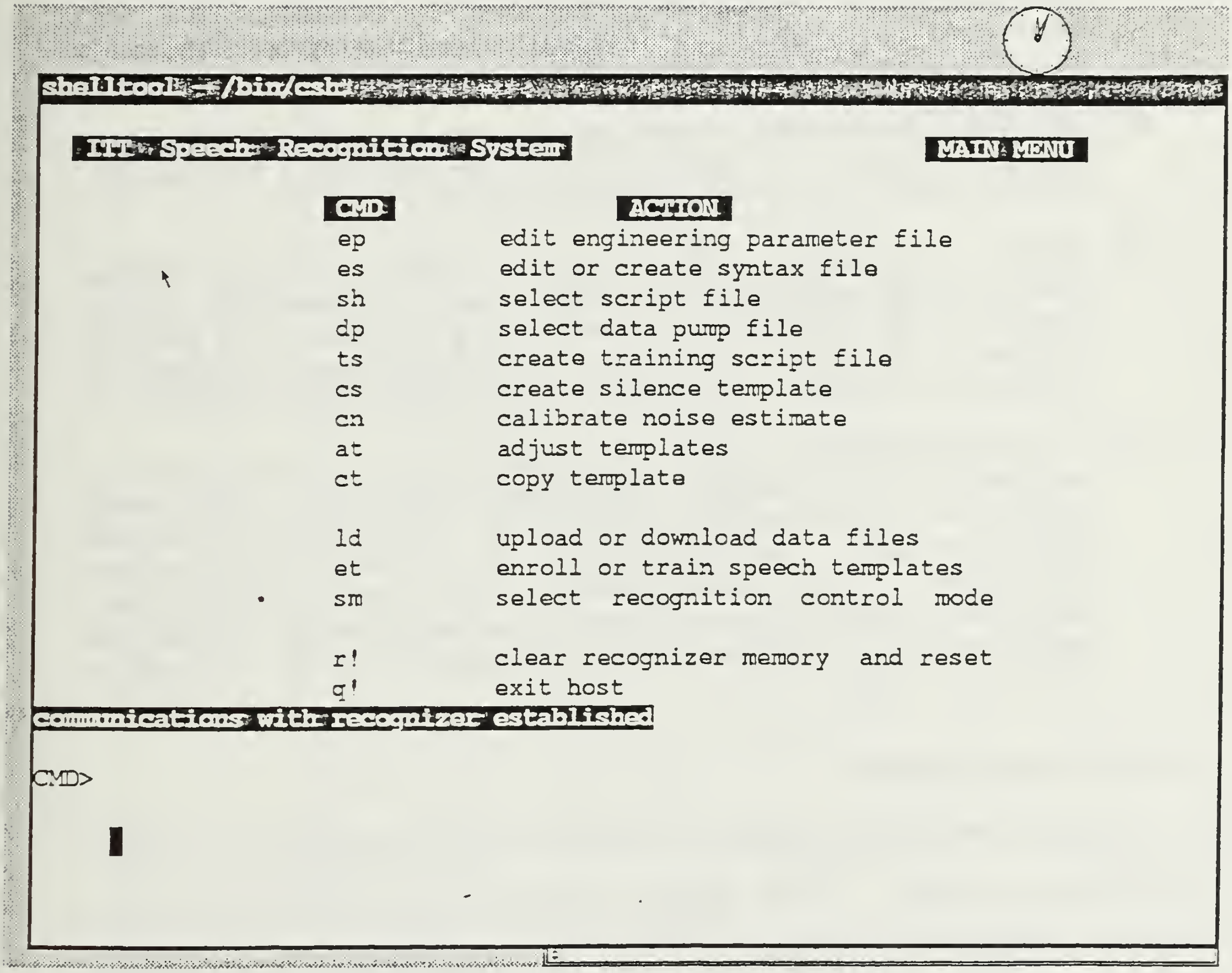


Figure 4.4

<u>Main Menu</u>

ep: The engineering parameter file (Table 4.2) contains a large number of system features which allow the system to be tailored to the application. This includes the adjustment of rejection threshold settings, pause lengths, and gain controls. While many of the parameters are at "factory" setting, tuning the board to optimize performance for a specific application may be required. Entering the "ep" command allows the user to view the file and adjust the current parameter settings, as needed.

cn: An ability to operate in a variety of noise conditions is a prerequisite for most voice applications. The ITT system allows for the calibration of the ambient noise by executing the "cn" command from the main menu. Calibration of the noise requires approximately 15 seconds.

at: According to the user's manual, templates should be created in quiet conditions. In order to adjust the templates for the calibrated noise, the "at" command is issued.

dt: Before a recognition session can commence, both the syntax and the user vocabulary templates must be successfully downloaded to the recognizer. Downloading templates (dt) may be aborted if the path is incorrect, if templates are corrupted or missing, or if there is a recognizer synchronization problem.

es: A syntax file may be either created or edited by issuing the "es" command. ITT software allows the creation of a node-based syntax, each node consisting of words which may be reached within the node. The editor allows the addition, deletion, and connection of nodes, as required, to create the desired syntax. Recognizer limitations include a maximum of 60 words per node in a total of 255 nodes. The maximum number of words is 400 [Ref. 17:p. 5].

D. SYNTAX DESIGN

Syntax design was based on the vocabulary necessary to operate the CATCC displays. A copy of the combined syntax supplied with the system is found in Appendix B, page 1. Total size of the working vocabulary is 71 words organized into 30 nodes. All three displays (Marshal, Approach, and Departure) can be supported by the syntax.

TABLE 4.2

ENGINEERING PARAMETER FILE

20000	path score rescaling threshold
2000	offset to calculate pruning threshold
6	max number of total options saved
1000	option pruning threshold offset
0	node number of starting node
1	end node number
1	weight assigned to downloaded templates
3	max weight allowed for templates
2	minimum number of training passes
3	max # times template length is adjusted
200	max delay allowed for results output
-1	penalty imposed for special loop back syntax node
-6	scale factor for relative gain term
0	window size for relative gain
6	programmable gain control in TMS320
2	scale factor for mel cepstral coef 1
2	scale factor for mel cepstral coef 2
2	scale factor for mel cepstral coef 3
2	scale factor for mel cepstral coef 4
2	scale factor for mel cepstral coef 5
2	scale factor for mel cepstral coef 6
2	scale factor for mel cepstral coef 7
2	scale factor for mel cepstral coef 8
32	offset for mel cepstral coef 1
32	offset for mel cepstral coef 2
32	offset for mel cepstral coef 3
32	offset for mel cepstral coef 4
32	offset for mel cepstral coef 5
32	offset for mel cepstral coef 6
32	offset for mel cepstral coef 7
32	offset for mel cepstral coef 8
1	log likelihood rejection enable flag
15	log likelihood rejection threshold
1	log likelyhood rejection filler training enable flag
0	noise tracker enable flag
0	noise tracker rejection enable flag
3	max # times a template can be updated per training session
0	DTW diagnostic loop forever enable flag
0	template warping function:
10	max length of pause nodes in special syntaxes
1	weight assigned to enrolled templates
0	data pump enable flag
-1	hardware push-to-talk flag
0	AGC enable flag
40	delay value before first gain increase
15	delay before each subsequent gain increase
4	Ebar noise tracker time constant (shift value)

Operating a display, however, requires only a subset of the combined syntax. Although not implemented, alternative syntaxes were considered by NOSC and are also found in Appendix B. These smaller syntaxes are designed to support exactly the specified function, thus eliminating syntactic overlap.

E. APPLICATION SOFTWARE OPERATION

1. Training the Recognizer

The ITT VRS 1280 is a speaker-dependent, connected speech system. Each speaker must initially train the vocabulary for his or her particular voice. This was accomplished by executing a NOSC-developed routine called "host." The initial screen, Figure 4.5, prompts the user for personal information. Figure 4.6 is the initial training menu displayed on the Sun workstation.

The first option allowed for enrolling and training of the digits 0 through 9. When executed, a series of ITT interface menus would be automatically executed (downloading templates, calibrating noise, etc.). After approximately 30 seconds, the user would be presented with the initial digits training screen found in Figure 4.7.

This screen is composed of two windows which are selected by moving the mouse-controlled cursor into the desired window. Training of the digits involved repeating the phrase or word immediately following the "PLEASE SAY... >" prompt. In this case, a base set of templates existed from which the user's utterance would be

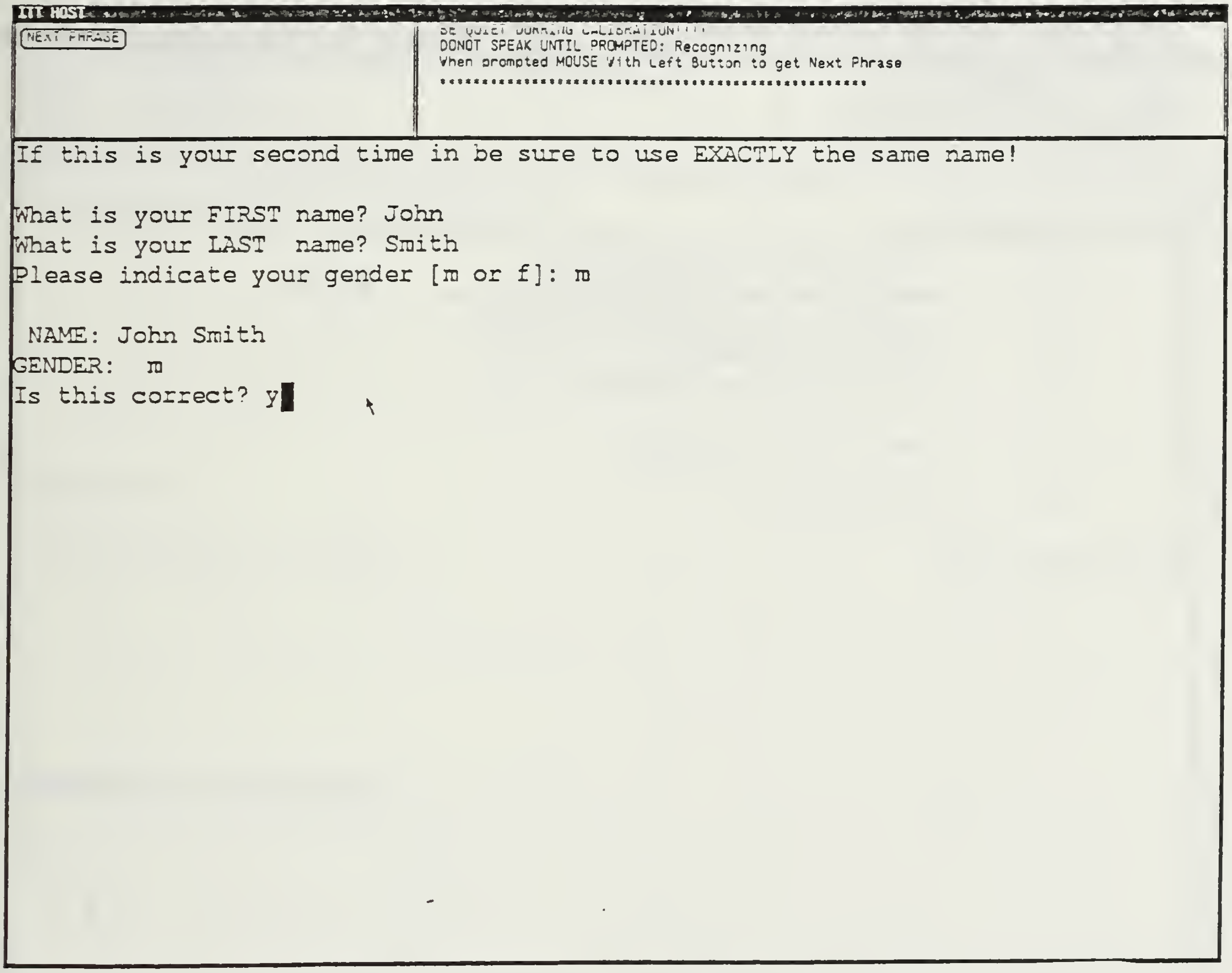


Figure 4.5

<u>Initial Screen</u>

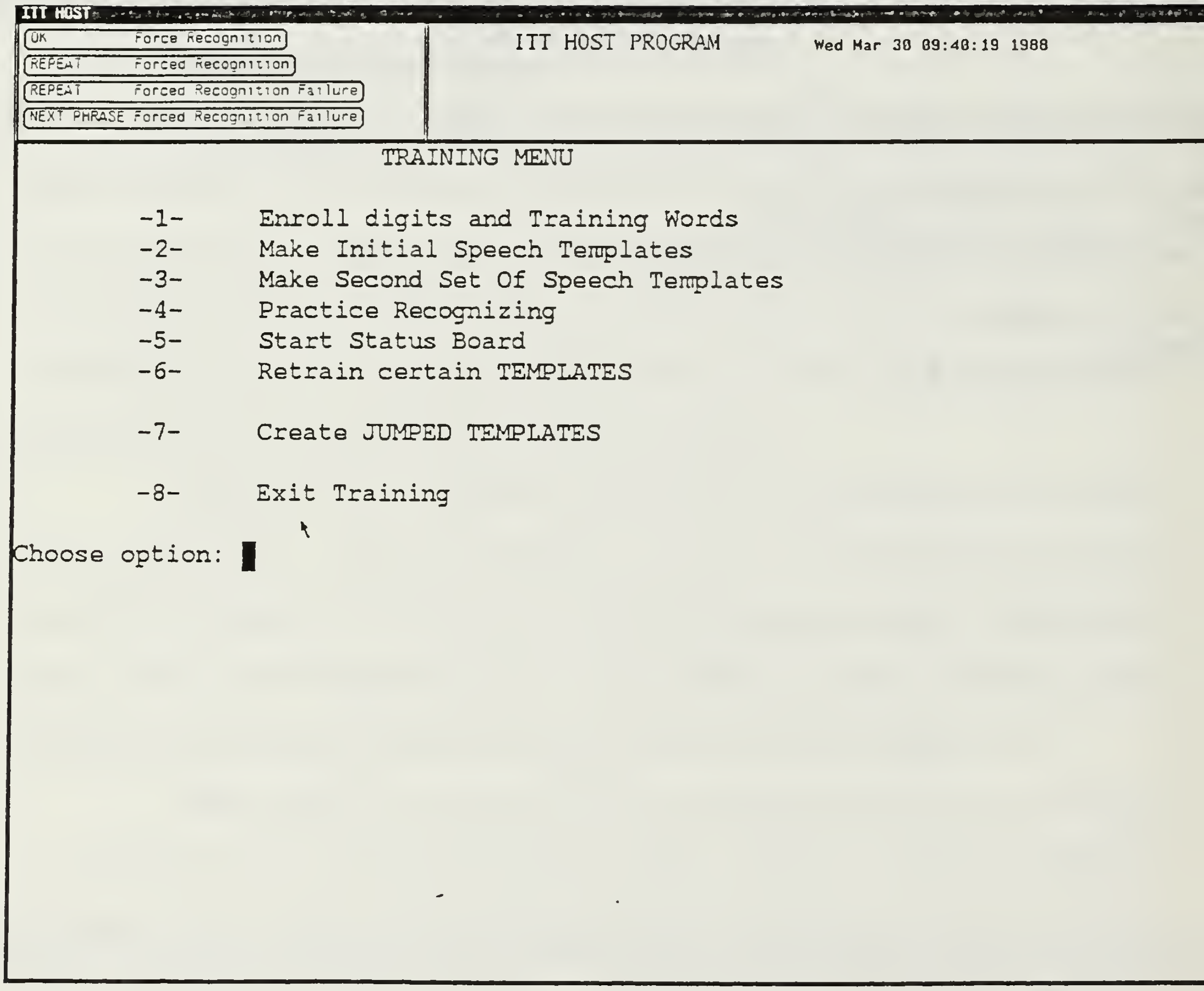


Figure 4.6

<u>Initial Training Menu</u>

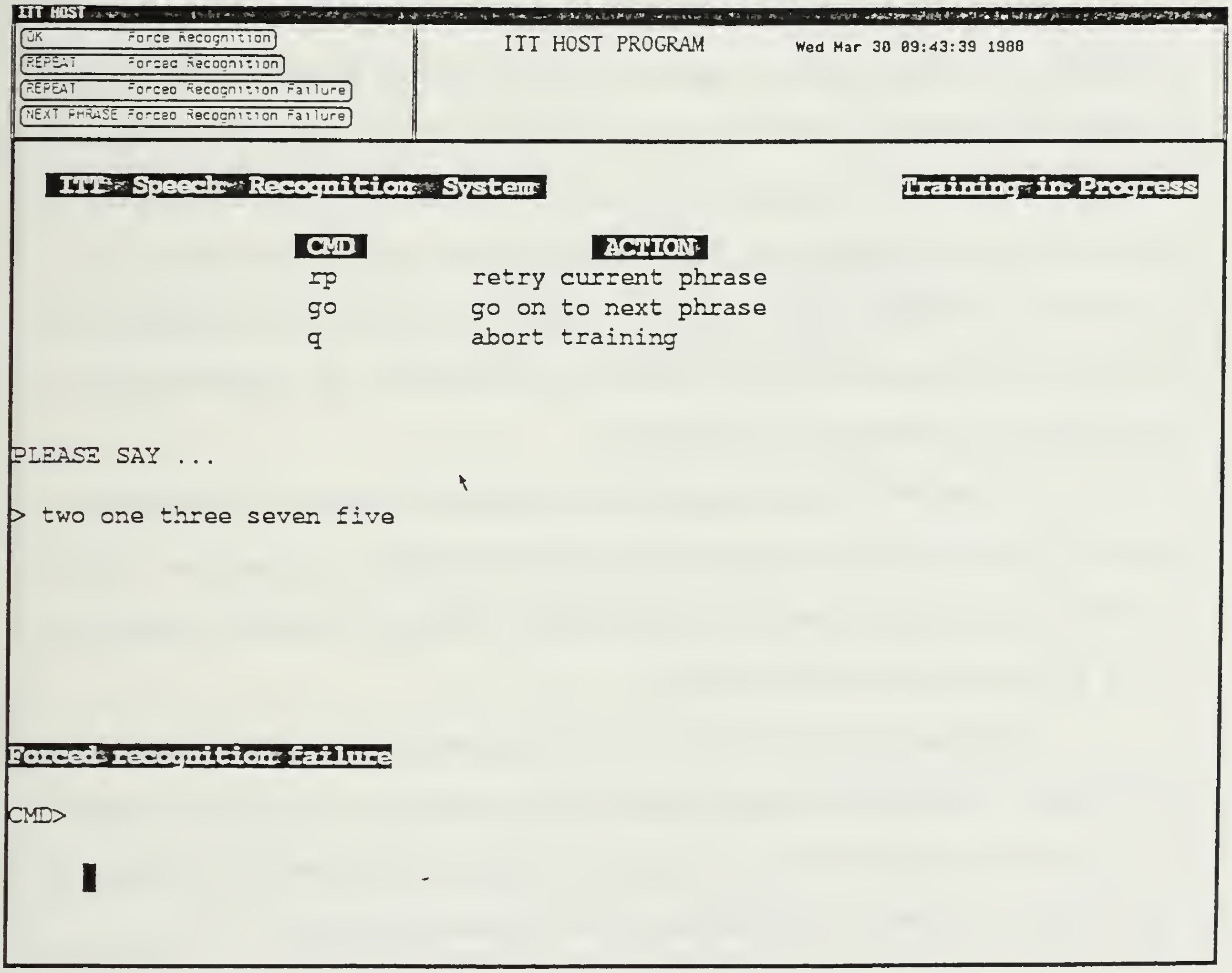


Figure 4.7

<u>Initial Digits Training Display</u>

bootstrapped. If the utterance was incorrect (generally a user word substitution error), a "Forced recognition failure" message would result.

If the utterance was recognized but significantly different than the base templates, a phrase recognition score would be displayed with the message "Forced recognition" (Figure 4.8). At this point, the user could either select the "REPEAT Forced Recognition" option and try the phrase again or the "OK Force Recognition" option, which required the recognizer to accept the input and force template adjustment. The degree of template adjustment is controlled through the Engineering Parameter File. Typically during the enrollment process, the template might be adjusted by 100 percent; as templates are adjusted during subsequent refinement processes, the adjustment factor might be reduced to 10 percent.

"Results: Open Recognition," shown in Figure 4.9, meant the user's utterance was recognized within specified parameters. As a result, the templates would automatically undergo adjustment and the next phrase would be presented.

Approximately three to five minutes were required to complete digit training for most individuals we trained. Users could exercise limited control over the system during this phase by executing one of the two-letter commands at the "CMD>" prompt.

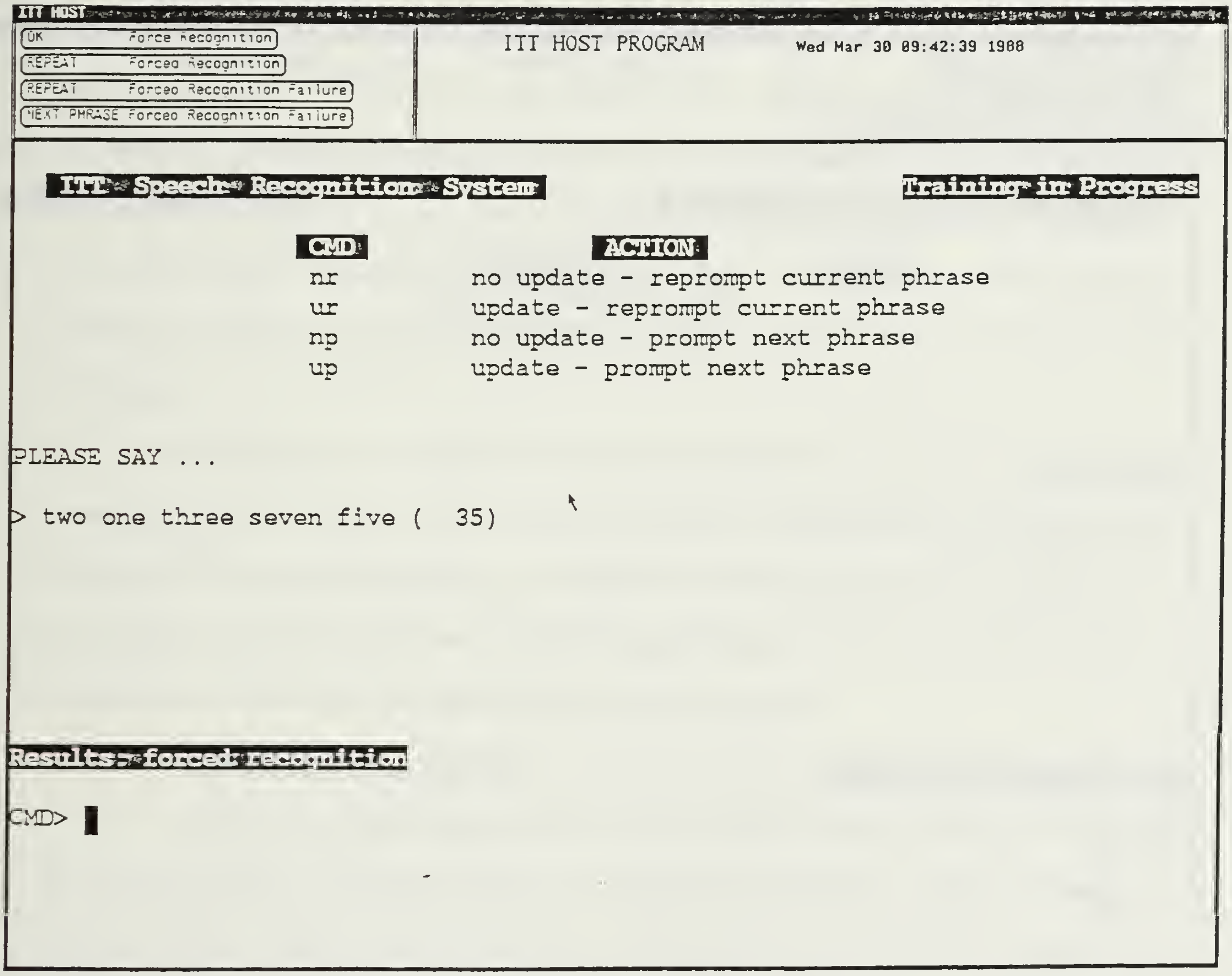


Figure 4.8

Forced Recognition Display

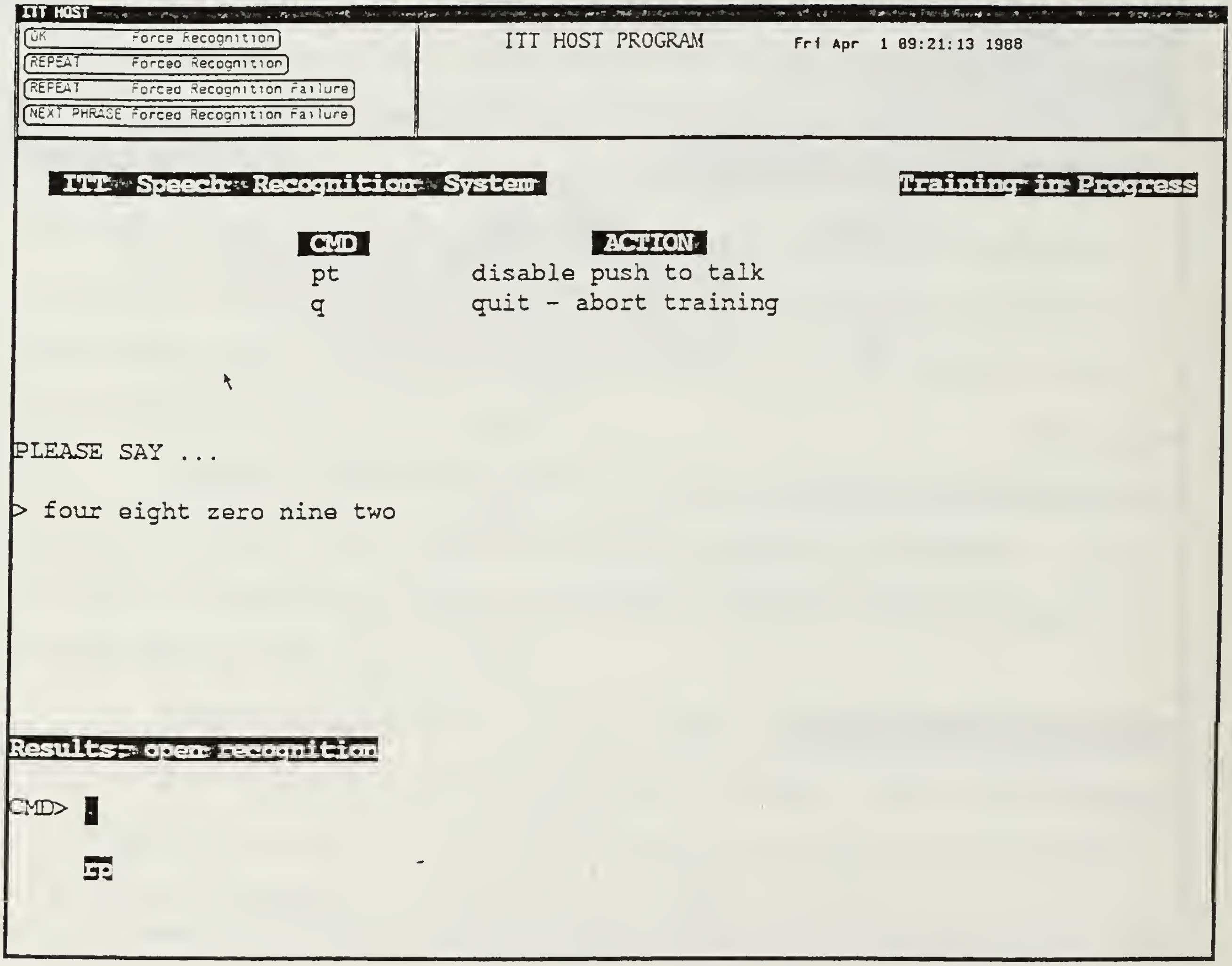


Figure 4.9

Open Recognition Display

Following initial digit training, option 2 on Figure 4.6 could be selected to create a set of templates for the application vocabulary. In this application, a pre-loaded set of vocabulary templates did not exist. Each template was created as the recognizer proceeded through a first pass of vocabulary words. During this phase of the enrollment process, speakers had to say each vocabulary word exactly as presented. Once enrolled, the vocabulary words would again be refined through ITT carrier phrases ("SAY airborne AGAIN") and in the actual syntax ("CHECK IN FUEL STATE THREE POINT ONE"). During this phase, the identical interface shown in Figures 4.7 through 4.9 was active. Enrollment time for the vocabulary varied widely between individuals; the average individual required approximately 45 minutes.

Although not used for the test, option 7 from the training menu allowed a user to train templates by bootstrapping from a set of previously trained templates. While this could reduce training time, the option was not used as the training method so that we could obtain templates without any possibility of previous bias.

2. Practice Recognition

Option 4 from Figure 4.6 allowed the user to practice using the vocabulary and the syntax. Following selection of the "Practice Recognizing" option, the user was presented with a screen shown in Figure 4.10. When the microphone was open, the recognizer would match signals against the vocabulary according to the syntax.

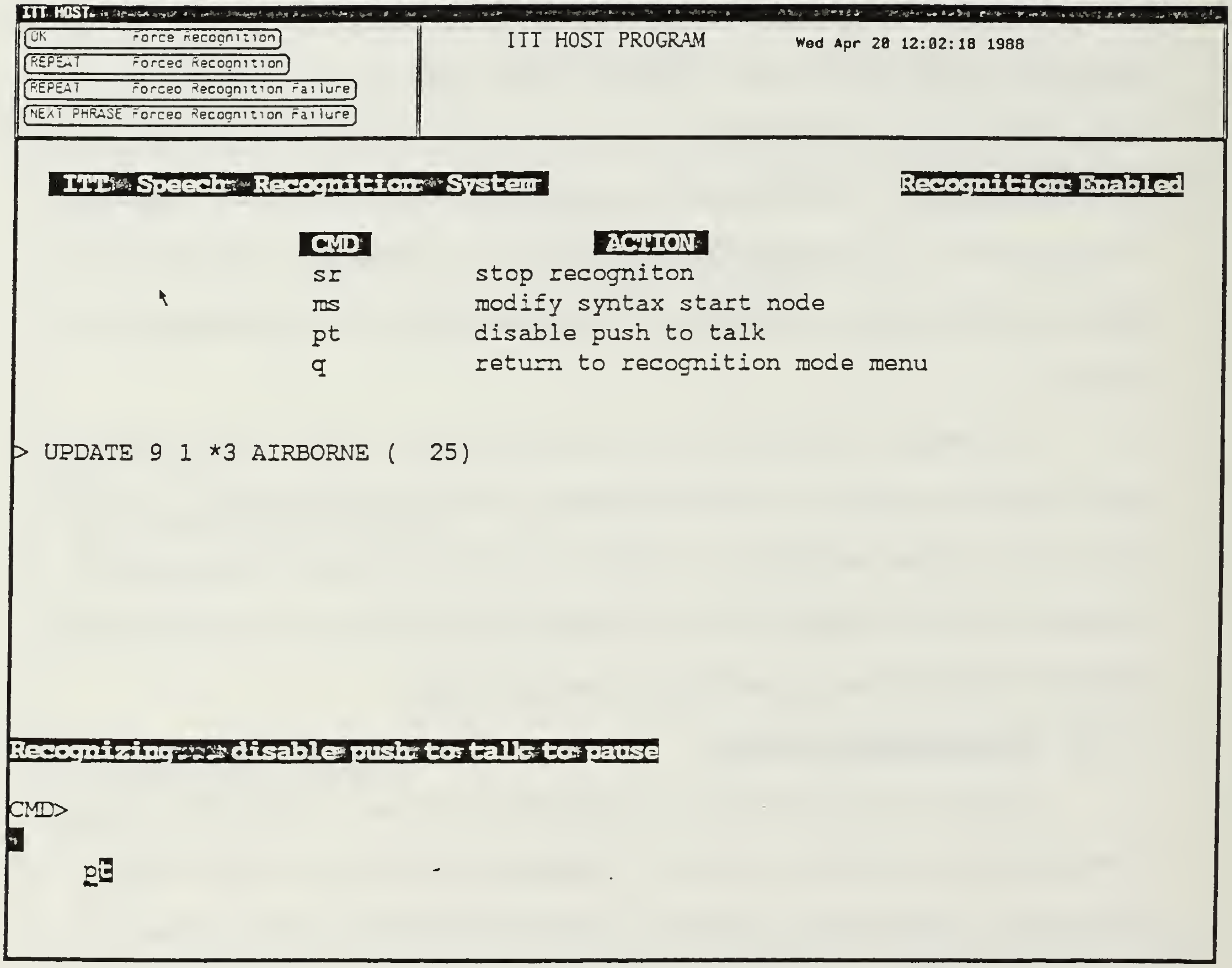


Figure 4.10

Practice Recognition Display

Following recognition, the phrase would be presented, accompanied by a phrase recognition score. If any words were not within the predetermined threshold they were marked with an asterisk. The session would end when the user typed a "q" to quit.

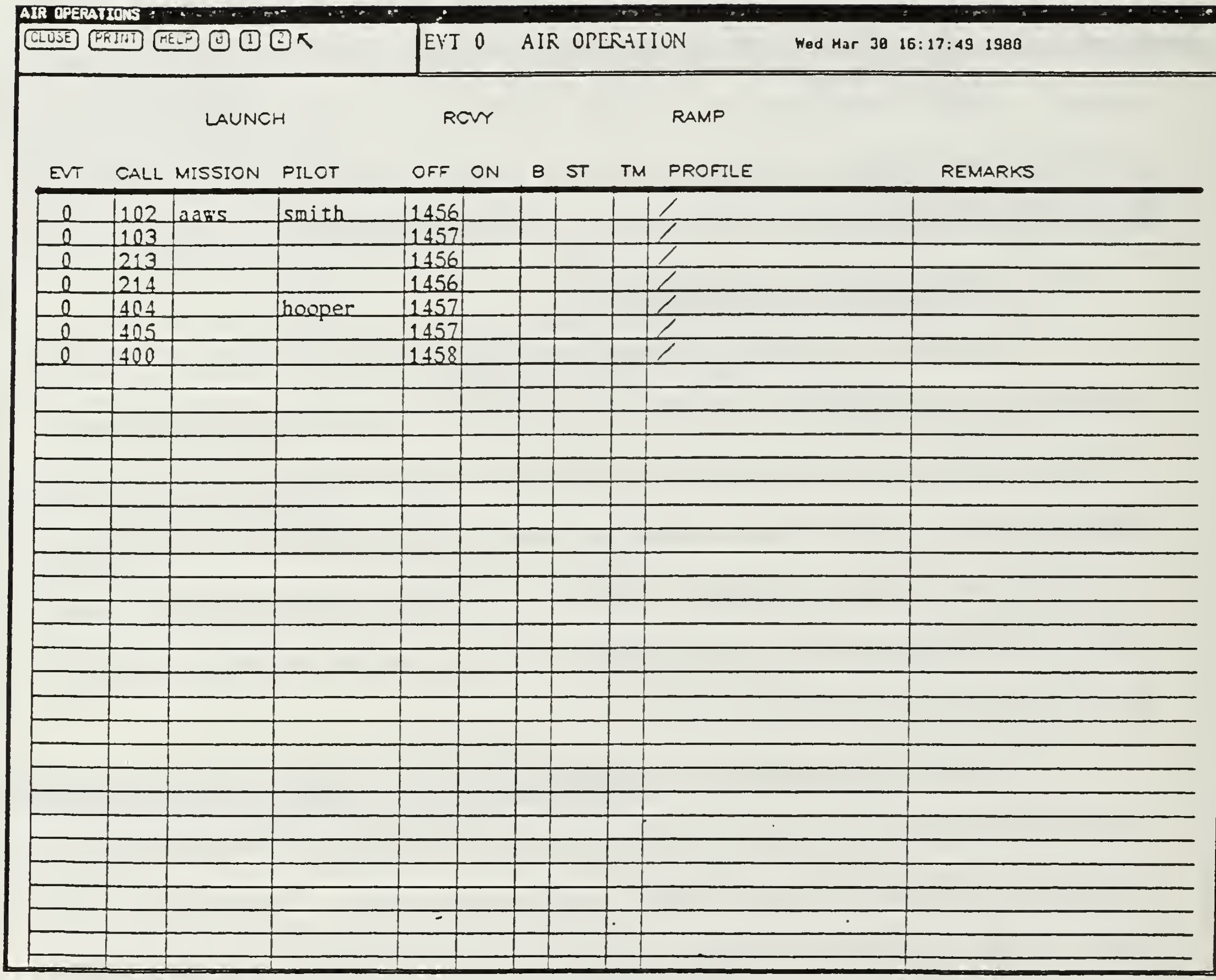
3. Retraining Templates

If specific templates were yielding inconsistent results, they could be retrained by exercising option 6 from the main training menu. When selected, the user would enter the word number requiring retraining. After recalibration, the word would be presented in two different phrases, which the user would repeat as before.

4. Operating the Displays

A series of visual displays designed to replace selected status boards was developed by NOSC. Input to the displays could be accomplished either via a combination of voice and keyboard entry or by keyboard entry only. When operating, each status board is displayed to a designated output terminal. The four displays supported are: Air Operation (Figure 4.11), Departure (Figure 4.12), Marshal (Figure 4.13), and Approach (Figure 4.14).

The Air Operation status board depicted in Figure 4.11 would have information entered via keyboard when the flight was anticipated. Included would be the pilot name and mission type. This data is not part of the syntax and thus would not be entered via the voice recognition system. As the flight departed, departure information, along with appropriate remarks, would automatically update the board.



AIR OPERATIONS

CLOSE PRINT HELP 0 1 2

EVT 0 AIR OPERATION Wed Mar 30 16:17:49 1988

		LAUNCH		RCVY					RAMP	
EVT	CALL	MISSION	PILOT	OFF	ON	B	ST	TM	PROFILE	REMARKS
0	102	aaws	smith	1456					/	
0	103			1457					/	
0	213			1456					/	
0	214			1456					/	
0	404		hooper	1457					/	
0	405			1457					/	
0	400			1458					/	

Figure 4.11

Air Operation Status Board Display

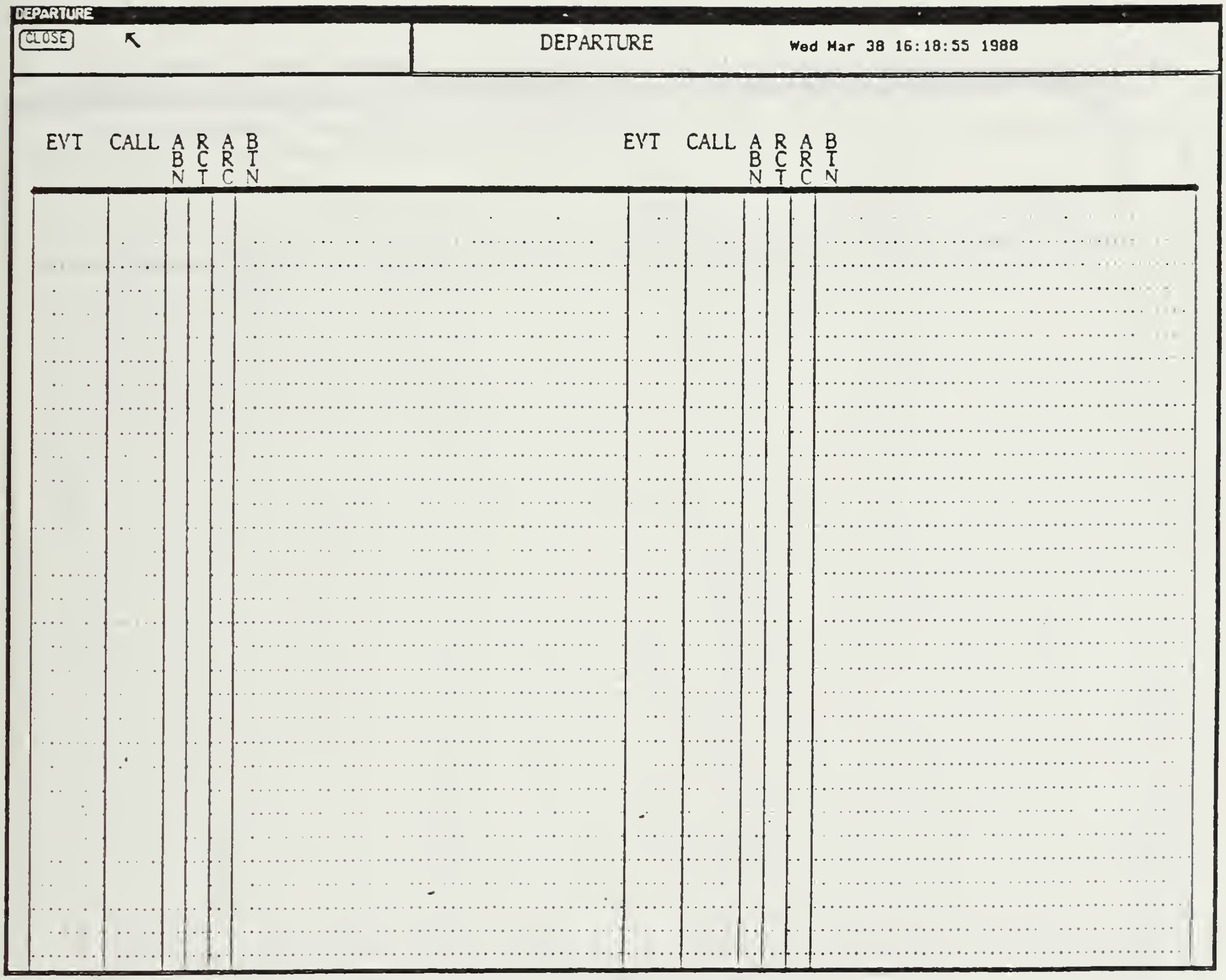


Figure 4.12

Departure Status Board Display

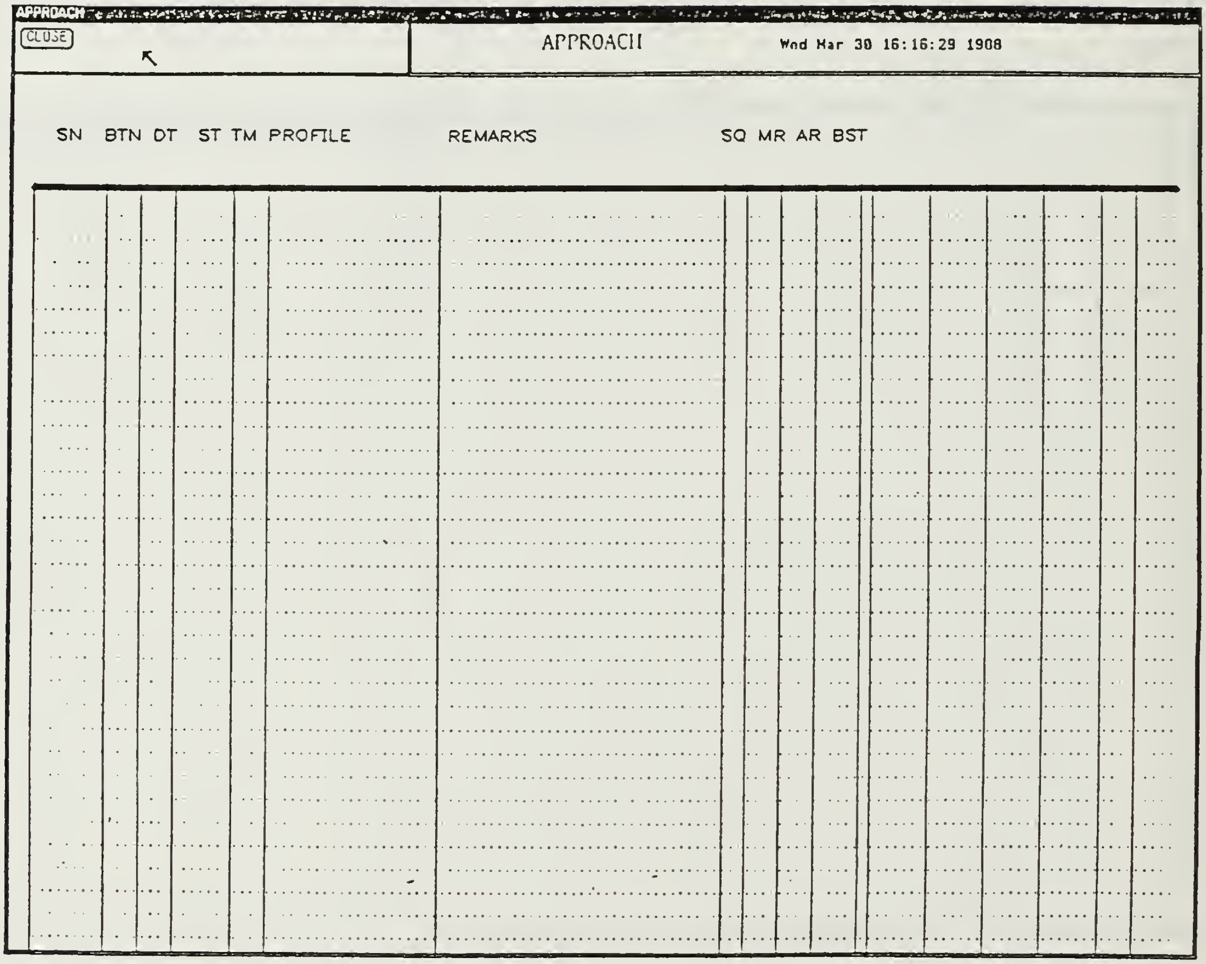


Figure 4.13

<u>Approach Status Board Display</u>

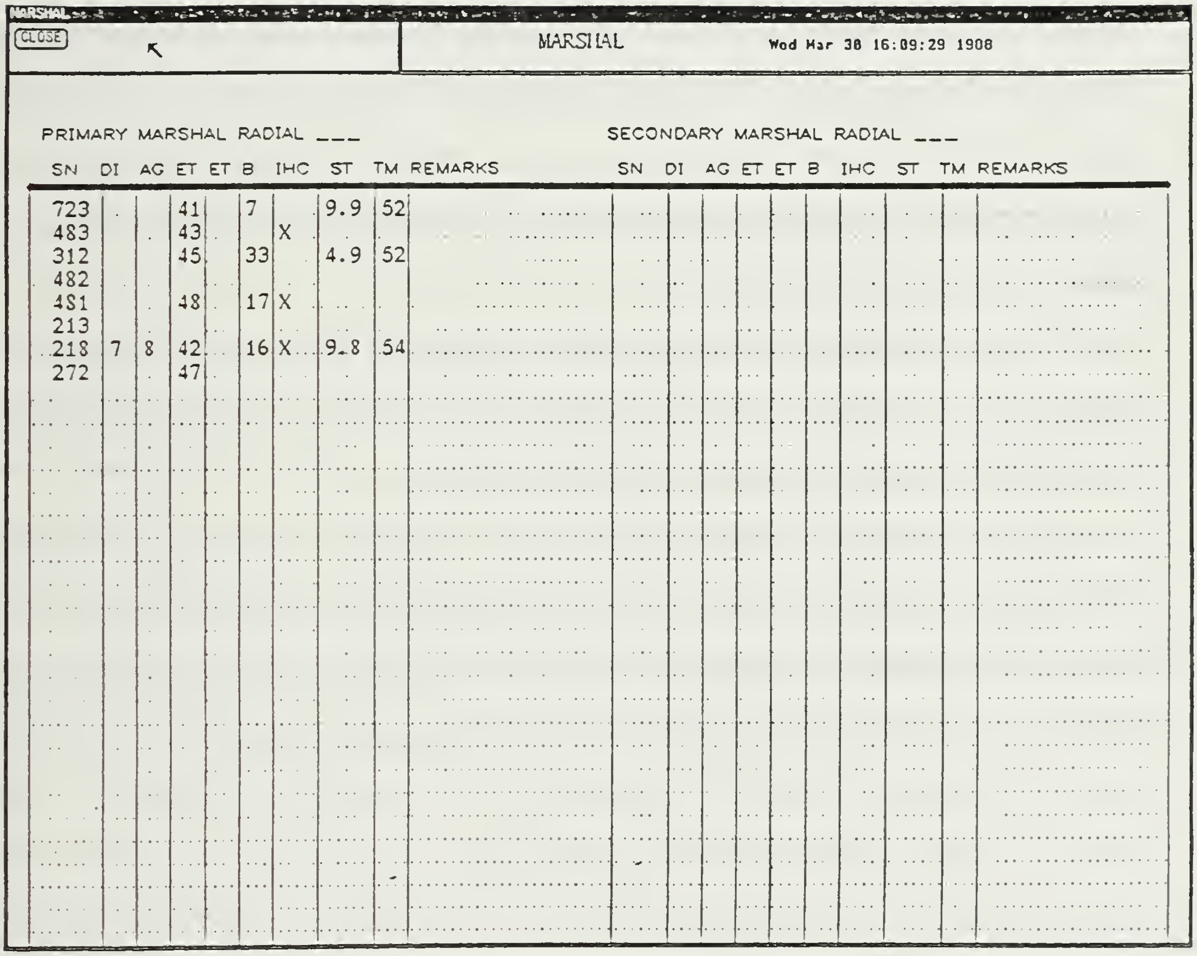


Figure 4.14

<u>Marshal Status Board Display</u>

Figure 4.12 represents the Departure status board. Again, the "talker" would enter the syntax supported departure information via voice (or keyboard). The event column would be filled in with the information available from the Air Operations status board.

Approaches are monitored by the Approach control board. Information that is monitored by this board would be used to automatically update the Air Operations board. A prime example is the aircraft "state" (or fuel status). As changes to the state are reported by the aircrew, it would be visually displayed on the Air Operations display once it is entered by the Approach "talker." Again, the operator has the ability to update his status board either via voice or manual keyboard entry.

The final status board available with the system is the Marshal display. Header information is not supported by the vocabulary and thus would be updated via keyboard entry.

The boards are maintained via the "UPDATE..." and "DELETE..." phrases. If the aircraft is deleted, all the information for that side number is removed and the display is automatically refreshed. Each operator maintains his own status board.

V. SYSTEM TESTING

The objective of this experiment was to evaluate the voice recognition accuracy of the ITT DCD Voice Recognizer/Synthesizer model 1280 VRS under four experimental conditions. Specifically, the experimenters' primary aims included evaluating the recognizer's performance under quiet (0 dBA) and noisy (75 dBA) environmental conditions as well as the relationship between the recognizer performance and the syntax utilized.

Two secondary objectives of the training and testing included an informal evaluation of the system's user interface and the overall training process. No particular experimental conditions were dedicated toward these ends; however, user surveys and extensive experimenters' notes on the approximately 200 laboratory man-hours were utilized to produce recommendations for further system development and training. These results, while principally anecdotal in nature, can at a minimum serve to guide final system designers toward the most productive designs based on the user interface and other human factors. Within this section, the only results pertinent to these secondary objectives can be found in the Questionnaire Results section. Additional comments regarding the overall user-friendliness of the system along with detailed recommendations on training have been deferred to Chapter VI for clarity.

A. DESIGN

A treatment-by-treatment by subject approach was utilized to test across the two noise levels and syntax conditions. A graphical representation of the design can be found in Figure 5.1. The subjects were considered a random factor and the syntactic and noise conditions were fixed.

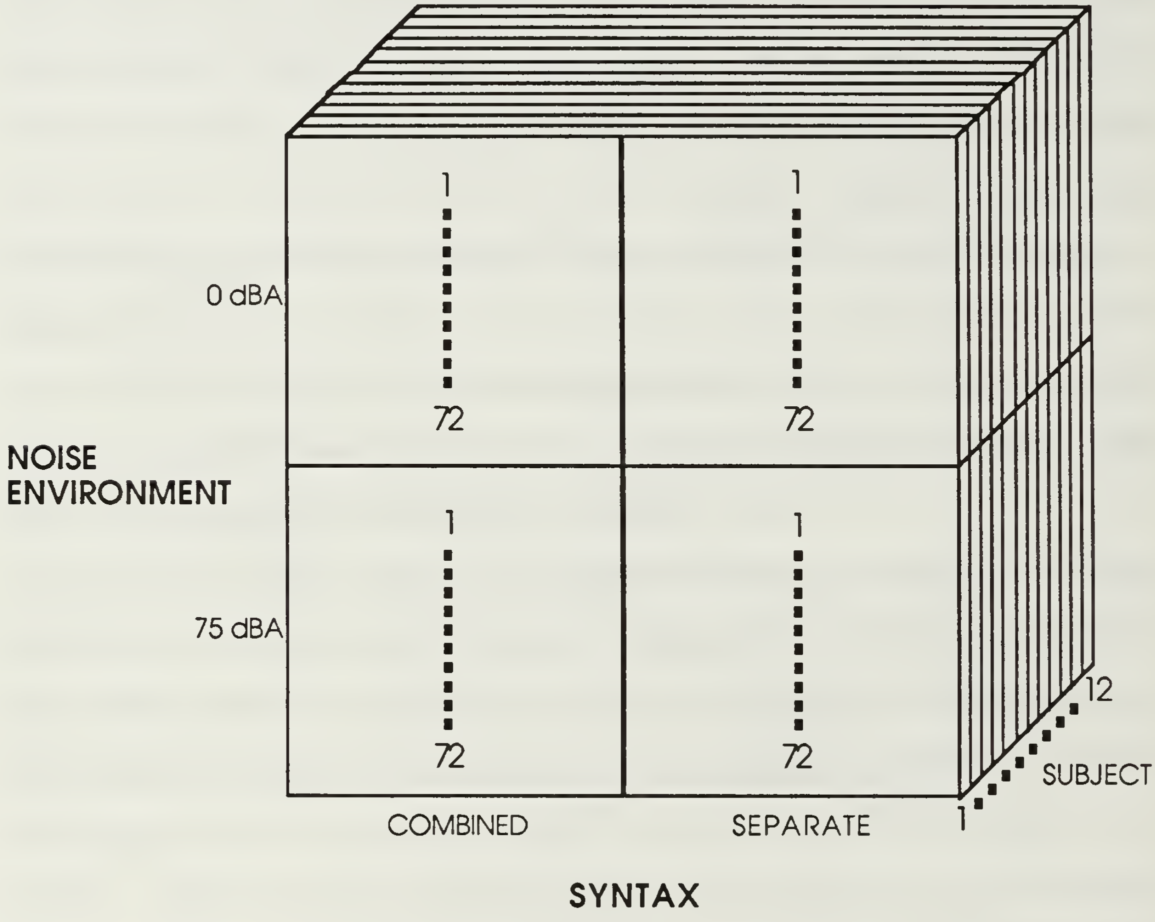


Figure 5.1

<u>Experimental Design</u>

At this point during experimentation, no attempt was made to simulate actual CATCC environmental conditions, control or otherwise,

beyond the use of selected CATCC phrases. This experiment was designed primarily to observe the relationship between noise level and recognition accuracy in order to determine possible limitations of the recognizer in the CATCC and to test the recognizer's sensitivity to the syntactic structure used for CATCC input.

B. SUBJECTS

Twelve volunteer subjects were recruited from the students at the Naval Postgraduate School. Because current DOD policy does not permit females aboard combat vessels, all subjects were male. Of the twelve subjects, nine were naval officers, two were U.S. Marines, and one subject was a DOD civilian. Six subjects had been exposed to a continuous automatic speech recognition system before and had between one and five hours of experience combined on discrete and continuous ASR systems. Eight of the subjects had direct CATCC experience, and eleven of the twelve had experience with the vocabulary through flight training/operations. In addition, all but one subject had extensive microphone experience in CATCC or other radio operations, as naval aviators, or as naval flight officers (navigators). Of the twelve subjects, six were from the computer systems management curriculum and six were from computer science. The level of subject service experience was reflected in ranks ranging from O-3 to O-4 in the Navy and O-3 in the Marine Corps. The civilian holds a GS-12 rating.

C. APPARATUS AND MATERIALS

A Sun-3/160M workstation with an ITT DCD model 1280 Voice Recognizer/Synthesizer was utilized for this study. The complete details of the system architecture can be found in Chapter 4, but it is worth noting here that the response time is reported to average .25 seconds with a vocabulary capacity of approximately 2,000 words [Ref. 16].

The Sun workstation and ITT ASR board were augmented with WYSE WY-60 terminals for prompts and recognition sets as well as a Shure SM12A microphone as an input device. A Hewlett-Packard model 465A amplifier was used between the microphone and ASR. The microphone was later changed to a Plantronics SNC 1436 noise-cancelling microphone, which connected directly to the recognizer board, allowing removal of the amplifier. These hardware changes were implemented prior to final testing and training and will be explained in the following section on training.

The Sun workstation components minus the computing unit itself, along with four WYSE terminals and the microphone, were all located in a 7' x 7' controlled Acoustical Environments chamber. The chamber is a nearly soundproof environment with internal noise registering 0 dBA when external noise averages 60 dBA. The noise for all stages was thus controlled, with noise induced through experimental conditions only.

Specific materials used in the conduct of the experiment included the following:

1. Graphical illustrations of the four syntaxes (i.e., Approach, Departure, Marshal, and Combined) for illustration of the syntaxes to the subjects (Appendix B)
2. A master instruction sheet for experimenters to insure uniformity in testing (Appendix C)
3. A test subject information sheet to gather basic subject information (e.g., name) and user interface and/or training problems or recommendations (Appendix D)
4. A training verification sheet for confirmation of subject vocabulary templates (Appendix E)
5. A subject-by-condition testing matrix (Appendix F)
6. Pre-testing instructions (subject) for the test (Appendix G)
7. Computer-loaded test files for each syntax (Appendix H)
8. A computer file of CATCC radio calls to use through a DECTalk voice synthesizer as part of the induced noise (Appendix I)
9. Response phrase sample file (Appendix J)
10. A post-test questionnaire to gather relevant subject information/qualifications and the subject's impressions of the system's usefulness (Appendix K)

D. PROCEDURES

1. Introduction

Before the conduct of the training or experimental sessions, a 15-minute introduction to the research was presented in a graduate-level course at the Naval Postgraduate School. During this introduction, the students were told the purpose of the research, what the experimental design was, and the approximate total time it would take to participate voluntarily. This was followed by a period for questions. It is worth re-emphasizing that the subjects did not receive monetary compensation or classroom credit for their participation which, as a

result, remained strictly voluntary. Subjects were asked to sign a roster indicating they were interested in participating and commit to three blocks of time, to include at least one two-hour block that would not impose on their school or personal schedules. These rosters were collected and a schedule was devised for the training and testing of 20 subjects, 18 of whose original time requests were able to be accommodated.

The experimental phase was originally divided into two sessions for each subject—training and testing. Both sessions were to be conducted in the Man/Machine Systems Design Laboratory at the Naval Postgraduate School inside the chamber previously discussed. All 20 volunteers were initially trained on the system in the manner described below, but numerous recognizer error messages and software bugs precluded the continuance of the testing phase. These difficulties were alleviated by telephonic and electronic mail consultations with NOSC designers/programmers as well as telephonic and on-site consultations with ITT technical representatives. The specific nature of the problems and solutions will be discussed in Chapter VI. As a result of the time lag experienced with these repairs, the number of subjects was reduced to 12 to allow completion of the testing within the fixed time constraint for the return of hardware to ITT and NOSC.

2. Training

Prior to the subject's arrival for a given experimental session, the experimenters would ensure that all equipment and forms were present. Appendix C was used to remind experimenters of various

training and testing procedures and ensure that training and testing of various subjects was consistent over time. A test subject information sheet (Appendix D) was filled out with the subject's name to record the time required for training and testing as well as any noteworthy difficulties encountered during testing or training.

Upon arrival at the Man/Machine Systems Design Laboratory for the experimental session, each subject was briefed on the training and testing methodologies and specific procedures they would be following. More specifically, the experimenter would first instruct the subject on using the speech recognition system. This included the following precepts:

- Position the microphone slightly to the side of and nearly touching the mouth.
- Keep microphone position constant during training and testing.
- Speak with consistent volume and speed.
- Speak in a style consistent with normal speech. Unusual enunciations were discouraged.

Next, the experimenter would brief the subject on the training to be conducted by introducing him to the graphical illustrations of the syntaxes of the words he would encounter as well as discussing the order in which the training would take place (i.e., digit training followed by full vocabulary). The subject was then told that following training he would be asked to read through a series of test phrases to ensure he had good-quality templates.

Once this introduction was completed, the subject would begin training the vocabulary words/phrases on the Sun workstation as

prompted by the system. After completion of the training passes, the experimenter would place the system into an open practice recognition mode and the subject was asked to read each of the phrases on the training verification sheet (Appendix E) three times. If any phrase was not completely and correctly recognized two out of three times, the experimenter would trace the recognition problem and retrain the template(s) for the word(s) until all phrases were recognized without error two out of three times. This ensured that quality templates for the various utterances were developed for each subject and allowed the subjects to visually see open recognition of their trained vocabulary.

3. Testing

Prior to explaining the testing procedure proper, two notes are in order here. First, the noise condition was set to 0 dBA or 75 dBA. The quiet condition was chosen in an effort to maximize potential recognizer performance. The loud 75 dBA condition was chosen based on the experimenters' familiarity with CIC environments and by actually manipulating the noise during experimental design to see what sounded loud yet would still be tolerated as a work environment. Thus this choice of a loudness threshold, while somewhat arbitrary, provides a basis for comparison when actual measurements of the CATCC noise levels can be taken. Such measurements were discussed but proved logistically beyond the capabilities of this research.

The second note to be made here relates to the syntax conditions. The two conditions are labelled "Combined" and "Separate."

Within the CATCC there are three stations which handle different types of statuses for the various aircraft. These are termed Approach, Departure, and Marshal. Each of these stations has its own syntax under the "Separate" syntactic condition; if an approach controller tried to use syntactic nodes (i.e., words and phrases) associated with Departure or Marshal, the recognizer theoretically couldn't find a response phrase match. This separation limits the number of word paths the recognizer must choose between to match the spoken phrase to a response phrase. In the "Combined" syntax, on the other hand, these three separate syntaxes are joined together so any of the personnel maintaining the status of the aircraft could use any of the vocabulary. For this experiment specifically, there are 72 text phrases, 24 from each syntax which can be tested through the syntax for which they were specifically designed or through a combined syntax. Thus, during a test of the "Combined" syntax, a subject would speak 24 Approach, 24 Departure, and 24 Marshal phrases. The recognizer would use a combined syntax in looking for the response phrases. During the "Separate" condition, each unique syntax would be used for those phrases normally used by the specific person updating the particular status. The overall question in the regard of syntax then is, "Is there a recognizer performance difference if the syntaxes are kept separate or can they be combined with no performance degradation?"

After training, the subject was given an explanation of the various trial conditions (Noise vs. Quiet environment and Combined vs.

Separate syntaxes), under which the recognizer would be tested. Appendix F is the subject-by-condition testing matrix developed to minimize any learning or proficiency biases. Subjects were told the order of the conditions in which they would participate and then given written pre-testing instructions (Appendix G) to ensure they knew what they would need to do to facilitate the testing. This basically entailed typing in the name of the test file of test phrases and reading them with the appropriate pauses to page down to the next phrases to be read when necessary. Most subjects reported being quite comfortable with this after doing one example prior to the beginning of testing. Notably, all subjects were Computer Technology (i.e., Information Systems) students, which resulted in little or no apprehension regarding their retrieval of the test files because this function is virtually routine in their studies.

The testing was then started with the subject facing one of the WYSE terminals with one screen of his first test file in his view. Test files for each syntax can be found in Appendix H. These files were generated at random with the exception that no syntactic path would be repeated until all paths were sampled at least once. The experimenter would establish the noise condition, if required, by calling the computer file containing simulated CATCC radio calls (Appendix I) and running these calls through the voice synthesizer. This noise was augmented by "white noise" produced by a standard portable radio tuned between broadcast frequencies. Noise was measured with a decibel meter prior to the subject beginning calibration

of the recognizer and, utilizing the same settings each time, averaged 75 dBA. The experimenter, regardless of noise condition, started a program to automatically record the recognizer's response phrases and set his screen to receive feedback on the subject's utterances. A sample of the response files created automatically can be found in Appendix J. The subject was then instructed to begin reading the phrases as per the instructions. Subjects thus had no feedback on recognizer performance utilizing the WYSE terminal, while the experimenter could watch the response phrases appear on the Sun workstation. In this way, the experimenter could "coach" the subject if he was speaking too rapidly or if he repeated a phrase or perhaps misspoke. Subjects were asked to reread any phrases they misspoke, whether discovered by the experimenter or self-reported. Each subject read through the various test phrases twice under each noise condition, once in a "separate" syntax and once in a "combined" syntax. Table 5.1 illustrates this more clearly.

After completing each condition, the subject was assisted with retrieving the next set of test phrases as required, the automatic response file was created for the next condition, and the subject began the next test phase.

After completing the final test condition, subjects were asked to fill out a survey (Appendix K) designed to gather subject data that might be pertinent to the recognizer's performance as well as the subject's impressions of the "friendliness" of the system and the

TABLE 5.1

TEST CONDITION PHRASES AND SYNTAXES

Phrase Number	Syntax	Condition	# of Phrases	Cumulative # of Phrases
1-24	Approach*	0 dBA	24	24
25-48	Departure*	0 dBA	24	48
49-72	Marshal*	0 dBA	24	72
1-72	Combined	0 dBA	72	144
1-24	Approach*	75 dBA	24	168
25-48	Departure*	75 dBA	24	172
49-72	Marshal*	75 dBA	24	196
1-72	Combined	75 dBA	72	288

*These three separate syntaxes with 24 test phrases combined to make up the separate syntax condition. The phrase numbers (1-72) in the separate syntax are the same phrases (1-72) in the combined syntax.

training itself. Subjects were then debriefed again on the purpose of the system being tested and thanked for their participation.

Finally, to ensure that data was not lost, a print-out of each subject's test response files was made and placed in a folder with the subject's questionnaire and subject information sheet. The contents of these folders were then held until scoring and results analysis began.

E. RESULTS

1. Dependent Variable

During all of the experimental trials, the response phrase of the recognizer was recorded automatically in response phrase

computer files like the example contained in Appendix J. The "correctness" of this response phrase as compared to the spoken phrase was the dependent variable for all trials. This dependent variable, however, was scored in two separate ways in order to examine the results from more than one perspective.

The work of Rodman, Joost, and Moody [Ref. 18] provided a method of scoring connected speech recognition systems utilizing reported phrases and the spoken phrases. This method provides two scores to each phrase spoken and was the first method chosen to evaluate the experimental response phrases. The first score in this method is based on the number of words reported correctly, in the correct order, divided by the number of words spoken. The latter is a calculation of the number of words reported incorrectly divided by the number of words spoken. This scoring method was utilized because of the number of types of errors that can occur in connected speech recognition. These include substitutions, insertions, deletions, merge errors, and split errors as well as preshadowing and postshadowing. Table 5.2 is provided (adapted from Rodman, et al.) as a brief introduction to these types of errors. Thus, this scoring method can provide more information in terms of the types of errors which are likely than simply recording the percentage of spoken phrases which were recognized without error.

The second scoring method utilized was, in fact, a method originally rejected as an oversimplification of the complex task of

TABLE 5.2

TABLE OF COMMON ERROR TYPES
OF CONNECTED RECOGNITION
(adapted from Rodman, et al., 1987)

Simple Substitution— One word is substituted for another.

e.g.	Spoken:	I can't fire faster.	
	Reported:	Tank can't fire faster.	score = <.75, .25>

Simple Insertion— An additional word is inserted.

e.g.	Spoken:	Coax fire on target	
	Reported:	Coax fire on target go.	score = <1.0, .25>

Simple Deletion— A word is left out.

e.g.	Spoken:	Coax fire on target	
	Reported:	Coax fire target.	score = <.75, 0.0>

Merge— Two or more words are recognized as one.

e.g.	Spoken:	Move tank slower right.	
	Reported:	Any slower right.	score = <.5, .25>

Split— One or more words are recognized as two or more.

e.g.	Spoken:	Can't go faster.	
	Reported:	Can't go fast gunner.	score = <.67, .67>

Preshadowing— A word resembling one of the syllables at the beginning of a correct word is inserted before the correct word.

e.g.	Spoken:	Move tank slower right.	
	Reported:	Move any tank slower right.	score = <1.0, .25>

Postshadowing— A word resembling one of the syllables at the end of a correct word is inserted after the correct word.

e.g.	Spoken:	M–60 turn rear.	
	Reported:	M–60 cease turn rear.	score = <1.0, .33>

measuring the recognizer's accuracy. It is the calculation of the percentage of response phrases which are equal to the spoken phrases without error. This method was utilized to illustrate the raw recognition rate in the prototype's environment where there was no method for error correction and where any required correction would necessitate repetition of the entire phrase. This, it is suggested by Pallet [Ref. 19], is the most appropriate method for an environment where this sort of whole phrase repetition is required for correction.

One final note on scoring is appropriate here. There were a few occasions during the sessions where the subject and the experimenter inadvertently missed speaking a phrase for one reason or another. These phrases were scored <-1, -1> across both scoring methods and were discarded during statistical analysis.

2. Results Using Rodman, et al. Scoring

Table 5.3 presents the analysis of variance for the first of the Rodman, et al. scores, that of the "number of words reported correctly (including being in the right order) divided by the number of words spoken." [Ref. 18:p. 272] That is, this analysis is fundamentally an analysis of the percent of correct words recognized. As illustrated, a significant main effect of syntax was discovered ($F = 4.7996$, $p < .06$) with no other main effects or interactions reaching a significant level. The overall mean score achieved by dividing the number of correct words recognized by the number spoken was .95958. This can be interpreted as indicating that nearly 96 percent of the words spoken

TABLE 5.3

ANALYSIS OF VARIANCE SUMMARY TABLE OF THE NUMBER OF WORDS REPORTED CORRECTLY DIVIDED BY THE NUMBER OF WORDS SPOKEN

SOURCE	df	SS	MS	F	p
Noise (N)	1	.0686	.0686	.4613	
Syntax (S)	1	1.1932	1.1932	4.7996	<.06
Subjects (Su)	11	4.5894	.4172		
N x S	1	.0602	.0602	.4343	
N x Su	11	1.6360	.1487		
S x Su	11	2.7343	.2486		
N x S x Su	11	1.5245	.1386		
Error	3387	70.7902	.0209		
TOTAL	3434				

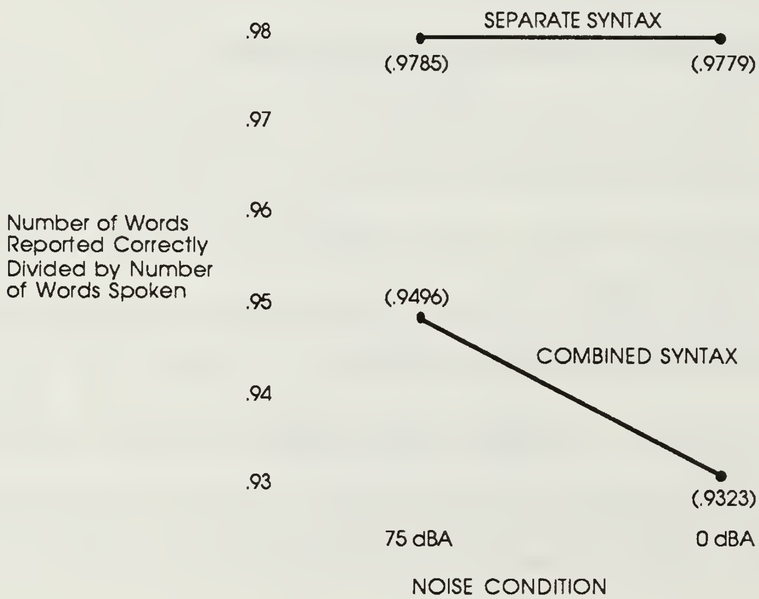


Figure 5.2

<u>**Syntax vs. Noise Correct Results Using Rodman et al. Scoring**</u>

are recognized correctly in the correct order. The mean scores for the number correct divided by the number spoken by syntax are shown in Figure 5.2.

Table 5.4 presents the analysis of variance for the second of the Rodman, et al. scores, that of the "number of words reported incorrectly divided by the number of words spoken." [Ref. 18, p. 272]. This analysis, therefore, is fundamentally an analysis of the percent of incorrect words recognized. In some cases, however, the number of words reported incorrectly can and does exceed the number of words spoken, thereby creating a value greater than one. Thus, in this sense this measure is not a strict percentage. As shown, a significant main effect of syntax was again discovered ($F = 5.1580$, $p < .05$) with no other main effects or interactions reaching a significant level. The overall mean for these calculations was .03930. Mean scores for the numbers of words reported incorrectly divided by the number spoken for each syntax are shown in Figure 5.3. The relatively low value of this score indicates that the errors of the system tested tend to be primarily deletion or substitution errors. This was found true by observation alone but these results can statistically provide the basis for recommendations concerning correction schemes which will maintain the portion of the phrase that is correct and insert or replace for the deletion or substitution as appropriate to produce the desired output.

TABLE 5.4

ANALYSIS OF VARIANCE SUMMARY TABLE OF THE NUMBER OF WORDS REPORTED INCORRECTLY DIVIDED BY THE NUMBER OF WORDS SPOKEN

SOURCE	df	SS	MS	F	p
Noise (N)	1	.0596	.0596	.3174	
Syntax (S)	1	1.5577	1.5577	5.1580	<.05
Subjects (Su)	11	5.2736	.4794		
N x S	1	.1027	.1027	.7431	
N x Su	11	2.0660	.1878		
S x Su	11	3.3218	.3020		
N x S x Su	11	1.5197	.1382		
Error	3387	91.6240	.0270		
TOTAL	3434				

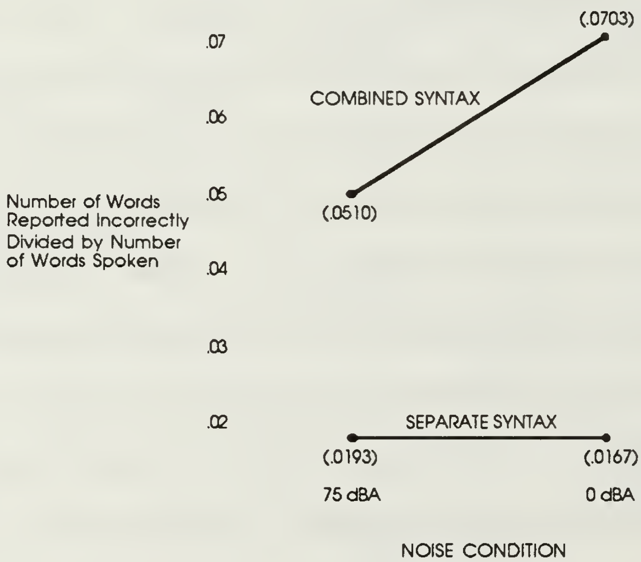


Figure 5.3

<u>**Syntax vs. Noise Incorrect Results Using Rodman et al. Scoring**</u>

3. Results Using Percentage of Phrases Recognized With and Without Error

Utilizing the second scoring method by simply figuring the percentage of phrases recognized with and without error created a distribution which was binomial rather than normal. Research has shown that the F test is very robust and can give an indication of significance despite this type of distribution [Ref. 20]. An analysis of variance was therefore conducted and yielded the same significant main effect of syntax as with the previously mentioned scoring methods. The results indicated an F value of 5.3920 with $p < .05$. Also similar to the other scoring method, no other main effects or interactions reached a significant level. The overall mean for correct phrases using this straight percentage scoring method was .90160, while the mean incorrect is, of course, the remaining .09840. These scores, while appearing lower in terms of recognition quality, are averaged across all four experimental conditions and ranged between 87 percent completely correct recognition in the noisy environment with the combined syntax to roughly 93 percent for the separate syntaxes under both noise conditions. It is clear that the separate syntaxes provide a statistically better likelihood of completely correct phrase recognition, as illustrated with this scoring method, and more completely correct phrase recognition when errors do exist, as shown with the first scoring method. This result, combined with an error correction scheme, may present a design modification which is not only statistically significant but practically significant. This notion will be discussed further in the following chapter.

4. Questionnaire Results

The user survey conducted was targeted specifically to determine the pertinent demographic information about the subjects (e.g., experience level, military grade) and their opinions regarding the training and the user interface of the prototype system. Questionnaire results indicated that 4 of the 12 subjects had no CATCC experience, 2 had been exposed to the environment indicating experience levels of 5 and 20 hours, and 6 of the 12 officers had an average experience level of 26.3 months through exposure in flight briefings or direct assignment. All subjects indicated they were very comfortable (9/12) or comfortable (3/12) with the vocabulary used in the experiment. In addition, 11 of 12 and 1 of 12 responded they were very comfortable and comfortable, respectively, with using a microphone.

Figures 5.4 through 5.7 indicate the subjects' responses to the training itself and the user interface the system provided through the hardware discussed earlier. As is graphically evident in Figure 5.4, all subjects found the training "Quite Easy" at the very least, and seven of them rated it the highest possible "Very Easy." The experimenters believe, however, that there is something of a subject/experimenter bias with the normal peer relationship existing between the two. That is, subjects may have felt that they were rating the quality of the experimenter as a trainer and were biased by their normal relationships. The intent of the question was not to measure this but rather to

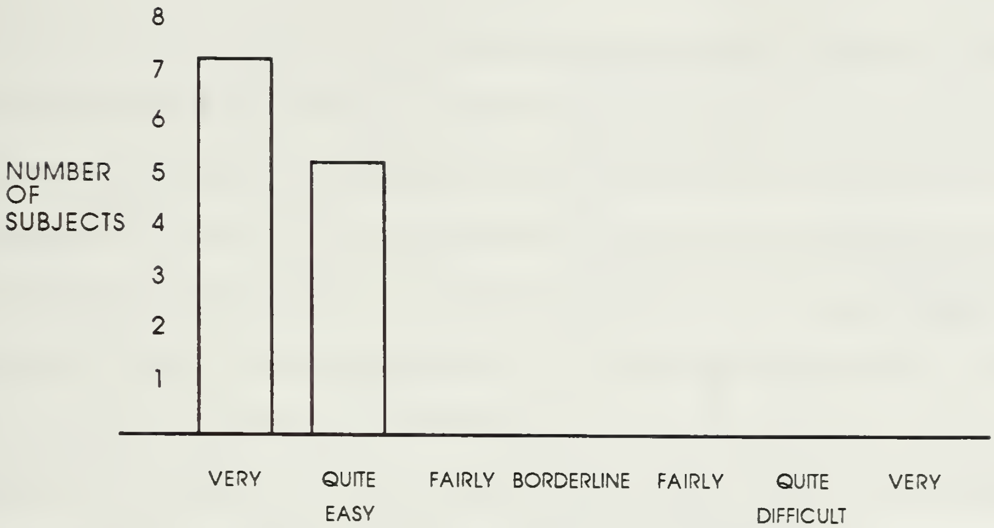


Figure 5.4

<u>The Training Session, as Guided by the Experimenter, Was:</u>

Figure 5.5

<u>The Quality of the Sun Workstation Display Used for Training Was:</u>

Figure 5.6

The Quality of the WYSE Display Used for Testing Was:

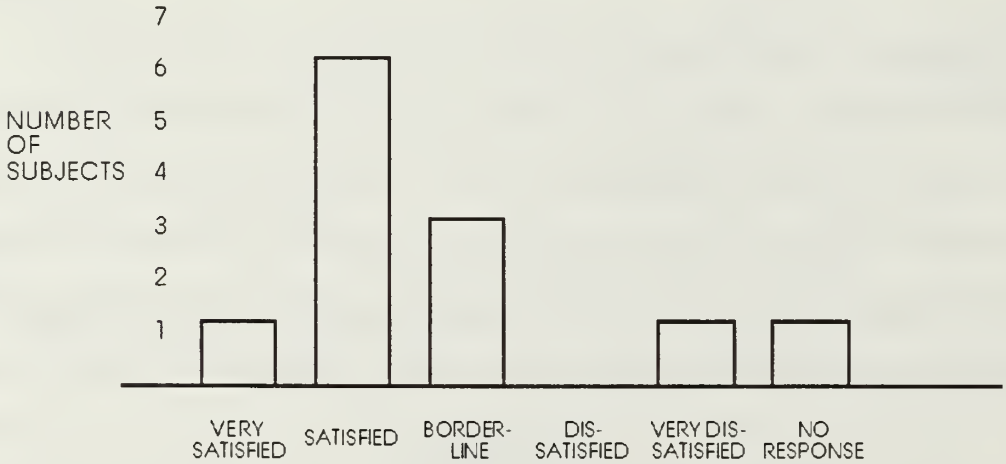


Figure 5.7

How Satisfied Were You With the Ergonomics of the Microphone Set?

get the subject's reaction to any delays experienced because of system hardware or software errors. These sorts of delays were recorded by the experimenter for all subjects during training and testing and will be commented upon in the conclusions and recommendations chapter. Figures 5.5 through 5.7 indicated subjects' reactions to various components of the interface between the user and the system. Most of the system will change prior to final implementation, as is the case with many prototypes. This is especially true of the visual displays since they will need to be readable from certain distances in the CATCC and therefore will need to be designed with the appropriate size, illumination, and/or colors. Subject opinions about the screens were relatively positive as per Figures 5.5 and 5.6, but the reactions to the microphone were the most varied. This points to a particularly critical design consideration because the microphone utilized was one of the few hardware components which eventually may be carried over into the final design.

The most critical questions addressed by the subjects were the final four. The first two questions were to elicit whether the subject felt that voice recognition technology was appropriate for the CATCC environment. The results of these questions can be found in Figures 5.8 and 5.9. The totals for each of the questions are somewhat misleading depending on the credibility we assign to those without experience or exposure to the CATCC environment. In fact, as illustrated by the figures, although those with experience or exposure

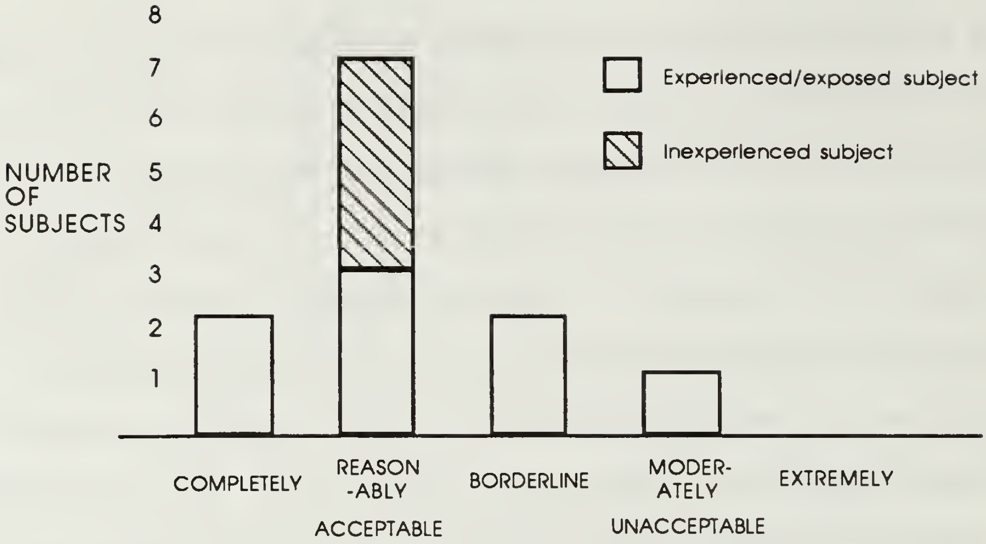


Figure 5.8

How Acceptable or Unacceptable Do You Feel Voice Input Technology is for the CATCC or CIC Environment?

Figure 5.9

If You Were Responsible for the Operation of a CATCC or CIC, How Would You Accept a Fully Developed Voice Input Status Board System to Replace the Current Methodology?

to the CATCC find voice input technology generally more acceptable than borderline or unacceptable for the CATCC or CIC environment, they are not as "positive" as are their less-experienced peers. The same can be said for their responses to whether they would accept a fully developed system if they were responsible for the operation of a CATCC. A listing of the responses to the final two "open-ended" questions can be found in Appendix L. Most responses center on the issues of reliability, maintainability, display quality, noise, and the trainability of the system. As will be further discussed in Chapter VI, these topics may become weighty considerations for final design features.

VI. EVALUATION, RECOMMENDATIONS, AND CONCLUSIONS

This chapter is a compilation of the experimental results and the recommendations and conclusions which logically follow. The first section is an evaluation of the NOSC prototype system. This is then followed by a section of recommendations to final system designers. These recommendations, while they may be linked to objective experimental results, may in fact be based on the results of user surveys (i.e., user experience) or the experimenters' own experience with the system. The recommendations are thus intended to be pragmatic and give a sense of what design elements might work and help eliminate potential problems vice those that are strictly proven by laboratory experimentation. The basis, whether experimental or otherwise, will be noted with each recommendation. These recommendations will then be followed by general conclusions.

A. EVALUATION

1. General

The prototype system provided for the evaluation of the use of speech in the CATCC environment evidenced at least one major flaw common to prototypes. Pressman points out that prototyping can be problematic as a model for software engineering because "The customer sees what appears to be a working version of the software, unaware that the prototype is held together 'with chewing gum and baling wire,' unaware that in the rush to get it working we haven't

considered overall software quality or long-term maintainability." [Ref. 21:p. 23] The NOSC prototype was typical of prototypes in this respect. The system as a whole was functional, but when experimentation began a number of dysfunctions occurred simply because the prototype, as a prototype, was not robust as a final system design would have been. For example, numerous recognizer errors were encountered. These errors, specifically error numbers 9 and 11, had been virtually unseen during NOSC development, but with the approximately 200 hours of training, testing, and simply "playing" with the system these errors were so abundant that they caused a week-long delay in final experimentation while an on-site consultation was conducted to fix the problems. Most of the additional difficulties discussed below are, in the opinion of the experimenters, related to this prototyping paradigm of development.

This is not to excuse these system flaws per se, but simply to evaluate them as a part of the environment in which the prototype was developed and the purpose (i.e., to evaluate the use of voice recognition technology in a CATCC) for which it was developed.

2. Hardware

At least one major hardware problem was encountered with the NOSC prototype. The delivered system utilized a Shure SM12A microphone connected through a Hewlett-Packard model 465A amplifier to the ITT automatic speech recognition board. A trace was attempted to isolate the source of numerous recognizer errors (averaging four to five per one-hour session), indicating lost

communication with the recognizer. Most attempts to eliminate other sources, such as software macro programs, were unsuccessful, but an ITT consultant pointed out that the button on the Shure microphone was not providing the hardware an on/off disconnect the recognizer board could identify. ITT provided a Plantronics SNC 1436 noise-cancelling microphone which supplied the connect/disconnect signal the recognizer required, which reduced the "lost communications with recognizer" messages to nearly zero. Notably, one other source, a software source, was discovered as related to the "lost communications with recognizer" errors. This will be discussed in the software section below.

Another concern, not necessarily a problem for prototype testing, is the long-term maintainability of the system hardware. Military systems are typically "ruggedized" to meet the unusually demanding requirements of 24-hour-per-day operational or combat environments. In fact, 27 percent of the user comments relating to the major issues with regard to utilizing voice input in the CATCC/CIC were tied to system maintenance, reliability, and system ruggedness (e.g., the ability to operate in degraded or unusual conditions). The system tested as a prototype appropriately used off-the-shelf commercial hardware. This hardware, while not put to the test in a closed laboratory environment, may have its ruggedness challenged with around-the-clock use in an operational or combat environment.

Hardware performance, other than the microphone difficulty, was quite positive. Objective experimental results put raw recognition

of correct words at nearly 98 percent, with incorrect words as low as 1.67 percent if separate syntaxes for the different stations are utilized. This recognition rate is commercially competitive with automatic speech recognition hardware and/or software and could, it is believed, depending upon how it is configured with software, prove quite effective for the system.

3. Software

A number of difficulties were encountered with regard to the software utilized in the NOSC system. The first, and initially the most harmful, trouble was a synchronization problem between the macro programs utilized to train the user's speech templates and the recognizer itself. The macro programs, written in UNIX Command Script, were essentially designed to run automatically once the training selection was made from the main menu. These programs would be automatically invoked at specific times during training. While the programs were loading and executing (usually less than a few seconds), the user would not have a prompt to speak so he would be silent. The recognizer, on the other hand, would be "looking" for an utterance. The recognizer would eventually "time out" prior to the completion of the macro execution, and by the time the user received his on-screen training prompt, a recognizer error would also be present. This type of software difficulty was addressed by Pressman as another problem with a prototyping methodology.

> The developer often makes implementation compromises in order to get a prototype working quickly. An inappropriate operating system or programming language may be used simply because it is

> available and known; an inefficient algorithm may be implemented simply to demonstrate capability. After a time, the developer may become familiar with these choices and forget the reasons why they were inappropriate. The less-than-ideal choice has now become an integral part of the system. [Ref. 21:p. 23]

As users of the system, it is unclear whether UNIX Command Script as a programming language is the optimal language for the system. The primary NOSC developer reports it was used based on his own programming background and familiarity. Suffice it to say here that a full-scale requirements analysis and subsequent design will be required utilizing the refinements discovered by working with the prototype. This lesson can be extended not only to this particular software aspect but also to the following software issues and the hardware problems previously discussed.

The user interface provided by the software was often *very problematic*. These problems fell generally into two categories: (1) features which are necessary for user-friendly system operation which are not implemented in the software, and (2) features which are built into the software which are in some way limiting to the user. An example of features which are not offered which would be necessary to make the system user friendly would be a volume meter so the user could adjust his voice volume to a level which will help create accurate templates. This approach has been used by other commercial vendors (e.g., Votan). Another example would be the ability to enroll single words vice the entire vocabulary. The current software requires the user to enroll the entire vocabulary for the CATCC at one time. This means that if the user makes a critical mistake enrolling one template

and wants to start "from scratch" for that word, he must re-enroll the entire vocabulary. This type of re-enrollment was required in nearly 40 percent of all subject training. An additional concern with regard to creating and adjusting templates was the sheer length of the training programs. Again, users could not break off their session without having to repeat what they had already trained, so any sort of incremental training was extremely limited by the software.

Yet another feature not found in the system software was clear and understandable terms for attempting to move within it. For example, to the user concerned with an operational environment, much of the voice recognition style language (e.g., "Open Recognition") could be transformed into more familiar terms (e.g., "OK"). These features, while minutiae to developers, can be the difference between a system that is truly geared toward the user and subsequently used by him/her and a system that is developed and shelved because users consider it unfriendly or difficult.

Current limiting factors of the software include such items as having to repeat an entire phrase to correct a single error and the inability to abort out of a training phase if one template is particularly poor without going through all well-trained templates upon returning. The first limiting factor is tied directly to the lack of a feature—that of an error-correcting scheme. Poock and Martin's research shows that error-correcting schemes have the potential to increase the efficiency of an automatic speech recognition system [Ref. 22], and the lack of such a scheme in this particular context requires the user to repeat

the whole phrase. For example, if a user said "UPDATE 1 0 5 PROFILE TRAP" and the recognized phrase was "UPDATE 1 0 9 PROFILE TRAP," with the present system software the user would simply have to repeat the whole phrase to get it correct before saying the word "SEND" to move the results to the appropriate CATCC status board. An error-correction scheme would allow the recognized "9" to be changed to a "5" without repeating the entire phrase, perhaps with an utterance like "CHANGE 9 to 5." This would increase user/system flexibility and maximize the recognizer's potential advantages (e.g., speed). Specific recommendations regarding a possible error-correction scheme will be detailed in the recommendations section which follows.

Aborting out of training and its subsequent retraining requirement is due to the types of recognition which are programmed as part of the software. The recognizer's message will be "OPEN RECOGNITION" if the phrase matches the template within the set recognition-scoring threshold. The message will be "FORCED RECOGNITION" if it falls within the next boundary of the thresholds; practically, this means the phrase was considered close but not within the bounds for open recognition. The utterance which elicited the "FORCED RECOGNITION" response may then be forced into the adjustment of the templates or repeated, depending on whether the user felt he uttered the phrase accurately or inaccurately, respectively. Finally, the user may get a message "FORCED RECOGNITION FAILURE." In most cases, this message means one of two things: The

user uttered the phrase so poorly or a different phrase altogether and the recognizer could not find a match, in which case repeating the phrase will remedy the problem, or the user uttered the phrase correctly and the template is poorly trained, thus the recognizer does not find a match. In this latter case, the user is trapped by the system software. He can delay the appearance of this phrase as a prompt by choosing "GO TO NEXT PHRASE" on the menu, but the phrase will reappear at a later time and eventually cause the user to abort out of training since he will not be able to achieve a match on this utterance. This combines with a previously mentioned feature which is not available on the menu, that is, to enroll or train a specific word or phrase, to make enrolling and training quite inflexible.

4. Syntax

Syntax accounted for an experimentally significant performance difference in the conduct of this evaluation. For the measure which divides the number of words reported correctly by the number of words spoken, recognizer performance was at nearly 98 percent for both noisy and quiet conditions utilizing separate syntaxes. The combined syntax, however, scored at near 95 percent and 93 percent, respectively, for noisy and quiet conditions. Similar results were obtained with the measure which divides the words reported incorrectly by the number of words spoken, combined/noise (.05), combined/quiet (.07), separate/noise (.019), and separate/quiet (.017). These results are statistically significant, at least at the PL .06 level, and, it is anticipated, would be practically significant for the CATCC

environment because of the need for accuracy in the operational environment and the expected volume of input during flight operations. Although an on-site evaluation of the CATCC requirements proved logistically impossible during the conduct of this research, it will be important for the final design effort to weigh the magnitude of input and the cost of errors against the cost of implementing the separate systems.

A final evaluation comment is in order here regarding the syntaxes utilized. Many of the subjects tested had direct CATCC or flight experience, the details of which are found in Chapter V. Nearly all of the subjects at one point or another commented about the inappropriateness of some aspect of the syntax. That is, subjects expressed such things as "You'd never say that" or "There's no such thing as ANGELS 90." It is believed that this is related again to the type of development (i.e., prototype) model used for this design, again illustrating Pressman's idea of the developer making "implementation compromises in order to get a prototype working quickly." [Ref. 21:p. 23] These concessions, while facilitating rapid development, can lead to less-than-optimal performance in a final system by providing more branches on a specific node than are actually legitimate real-world choices. It is of paramount importance that these syntactic settlements incorporated into the prototype model not be overlooked here or forgotten during final product development. A careful analysis and design of the actual syntactic rules of the CATCC operators should preclude errors caused by unnecessary nodal branching.

5. Other User Interface and General Evaluation Concerns

The current training interface for the user is fixed. He progresses through the series of menu choices described in Chapter IV and subsequently will have a set of voice-recognition templates on file for his use. This *may be* an appropriate training methodology for this environment and technology combination, but in practice the experimenters found a combination of pre-training with the vocabulary words/phrases and demonstration proved very effective, that is, it required less training session restarts. Whether this effectiveness is directly related to the training method is unclear because of the lack of flexibility of the enrollment and training of templates. For example, if we could start and stop enrollment at any location or just go back and re-enroll one word, would we need to pre-teach or demonstrate? Perhaps not, but this illustration makes clear the importance of carefully analyzing the training method to be utilized with the final system to provide a link between the new user and the system.

Once again recalling the issue of the use of the prototype as a design paradigm, we should emphasize the need for human factors requirements analysis. The prototype as tested received generally high marks from users when questioned about the quality or ergonomics of the work station, display, or microphones utilized. This could be anticipated in a laboratory environment for numerous reasons, including the following:

1. Subjects that are not actual users and are unaware of potential pitfalls in the human/system interface.

2. Lack of an operational environment to provide an accurate backdrop for the system operation.

3. Lack of realism associated with laboratory experimentation, especially when attempting to duplicate complex (e.g., shipboard) environments.

This is problematic, however, for attempting to generalize to the actual operational environment of the CATCC because the key human interface factors identified by researchers such as Monk [Ref. 23] are not the same across the laboratory and operational environments. These factors are the user population, the user task, and the user environment. The user population, that is, sailors from an aircraft carrier, could be utilized in laboratory experiments even though this was logistically impractical for the present work. This would narrow the human factors consideration to the user task and user environment, which still remain formidable human factor challenges. Examples of some of the numerous design techniques to be considered include:

1. What types of *control devices* are most appropriate for the task and environment? (e.g., mouse, joystick, foot-feed to control the voice input on/off switch)

2. What types of *software display devices* are appropriate to user output? (e.g., If the user needs symbols, how should they be displayed? How large should they be? What colors should the displays use? Are there any domain-specific colors/symbols that should be included/avoided?)

3. What types of *software control* should be available? (e.g., should the user be forced through menus or will commands be available for higher performance?)

4. What types of *hardware display devices* should be utilized? (e.g., raster scan displays, liquid crystal displays, plasma panels, printers, or even voice advisories through voice synthesis)

The detailed, all-encompassing scope of these considerations, while beyond that of this thesis, cannot be underestimated. Users which range from new trainees to those with hours of experience, combined with a task that can be extremely fast-moving but which requires accuracy in an environment which may be low in light but high in stress, noise, and concurrently required input tasks, create a nearly herculean task for the analyst striving to optimize the user system interface. But the human interface requirements analysis and subsequent input for overall design may determine whether voice input can be useful in the CATCC environment.

One final evaluative comment is in order here. As was previously discussed, a number of weeks were spent with software and minor hardware problems and finally remedied with consultations between the experimenters, NOSC developers, and ITT hardware experts. These types of difficulties could be effectively coped with during laboratory work because of its static nature. These same sorts of aggravation would render the entire system virtually worthless in the operational environment of a CATCC. Many of the test subjects experienced the errors and system crashes during initial training and subjects with CATCC experience reflected a healthy degree of skepticism regarding whether voice input technology was appropriate for the CATCC (Figure 5.8) and whether they themselves would accept a fully developed voice input status board system (Figure 5.9). Constructively, then, we must say that the present prototype system *is not ready for shipboard presentation*, even if it is merely used as the

requirements analysis tool it basically is. As it was utilized, there were too many errors still within the system for it to be used as an effective method of helping analyze the CATCC requirements. In addition, there are considerations with regard to creating a "negative" impression of the technology in an environment where the status quo methodology is so deeply rooted in naval carrier tradition. However, some subset of the prototype, or a more completely developed prototype, must be tested aboard ship in the operational environment to meet the requirements analysis to the fullest. The current prototype is simply not ready. Recommendations concerning this type of testing are contained in the following section.

B. RECOMMENDATIONS

1. General

Prototype testing requires a minimum level of functionality prior to system field testing. Any system which exhibits unpredictable and anomalous behavior cannot be adequately or fairly evaluated. With that concept, we are separating our recommendations into two distinct categories: short- and long-term recommendations. Our short-term recommendations are those deficiencies that must be solved prior to shipboard testing of the prototype. Issues or problems that must be considered prior to full-scale development, but which are not considered essential to the evaluation of the prototype, are found in our long-term recommendations.

2. Hardware Recommendations

During our evaluation, the hardware components (processor, displays, and ITT VRS 1280 recognizer) all performed without a single hardware failure. However, there are several short-term recommendations regarding implementation of the current suite of hardware. They are:

1. Incorporate manufacturer's recommended microphone system. The microphone system was our initial problem. However, using the Plantronics microphone, recommended by ITT representatives, we were able to correctly communicate with the hardware.

2. Operate/evaluate the complete prototype. To date, the system has only been evaluated in a scaled-down version of the full prototype destined for the ship. We strongly recommend that prior to field-testing, all three recognizers with a full complement of displays and input devices be installed and fully tested, as originally designed.

3. Acquire, test, and implement a large panel display. A major component of the system will be the displays used locally in the CATCC and those used as remote repeaters throughout the ship. Prior to at-sea prototype testing, we recommend implementation of a prototype large screen flat-panel display legible at a distance of several feet in low-lit conditions. Successful demonstration and validation of the concept is dependent upon successful incorporation of at least a prototype flat panel display.

4. Develop a shipboard cabling and power distribution plan. The cramped CATCC spaces require development of a detailed cabling plan prior to installation. Development of such a plan will avoid on-site wiring problems. The plan should map power outlet sources required to those available, and the specific location of cabling runs.

5. Test and implement a remote display. One of the major advantages of the system is the ability to display CATCC information remotely, thereby eliminating the human network of sailors. We recommend that this capability be fully tested and implemented, using flat panel display technologies, during the shipboard testing.

6. Develop a hardware performance limitation baseline. In the course of evaluating this prototype, specific performance criteria should be developed, tested, and documented. For example, what are the expected error rates with respect to various noise levels? Above what level of noise will the recognizer fail to recognize speech? Another criterion will be the response time under a variety of loading conditions. The primary concern here is to determine at what level of operation the components become saturated and to what degree the performance degrades.

Long-term recommendations associated with the hardware are more concerned with looking beyond the prototype. Of primary concern is that the prototype does not dictate the ultimate hardware (make and model) or the overall architecture to be employed. What is important over the long term are such hardware related issues as performance, maintainability, and reliability of the system. Accordingly, we make the following long-term hardware recommendations:

1. Consider alternative architectures. There are serious limitations associated with the current architecture. The prototype, as implemented, has a single point of failure. That is, if the Sun processor is no longer operative, the entire system is rendered inoperative. If this were to occur while deployed, the components would become unwanted baggage in the cramped CATCC spaces until the system could be repaired. In addition, there is no storage redundancy. All programs, voice templates, and systems software is stored on a single disk. Disk failure caused by vibration or dust (not unlikely in the carrier environment) would result in complete loss of data and voice templates (except for information archived on alternate media). A distributed network architecture based on stand-alone personal computers, each equipped with a recognizer and sufficient storage for voice templates, might be superior to the single processor system found in the current prototype.

2. Consider alternative equipment. The prototype system has several immediate disadvantages. Component size (large footprint), availability of maintenance while deployed, and lack of ruggedization are all long-term issues that must ultimately be addressed. Any system developed for Navy-wide use must include these issues in the system specification.

3. Optimize the input interface. Some combination of voice input and keyboard/pointing device will optimize the man-machine interface. Considerable effort should be devoted to identifying the best combination of input modalities.

3. Software Recommendations

Unlike the hardware components, operation of the software was not without problems. The short-term software recommendations are generally deficiencies that must be corrected prior to operation by CATCC personnel. Our long-term recommendations are not critical for concept demonstration but will become important during full-scale development. Recognizing the pre-production nature of the ITT recognizer, our recommendations will not distinguish between recognizer software problems and those problems caused by software developed by NOSC. Instead, we will recognize the problems in a generic sense, leaving resolution to some combination of improved ITT and NOSC software. Our short-term recommendations include the following:

1. Eliminate unpredictable operation. Included in this category are the recognizer errors previously identified. The system must not be installed in the CATCC without resolution of the various recognizer and lost communications errors.

2. Improve the training interface. The present training system is inadequate for the task. The inability to easily retrain/re-enroll selected words is considered a significant deficiency. The operator should be allowed to, at any time, retrain or re-enroll a word with a minimum of user command input. In addition, the operator should be allowed to discontinue an enrollment session without having to re-start the enrollment process. Finally, the user should be able to practice enrolling prior to actually creating voice templates. We recommend that the enrollment process be simplified, requiring at most one hour to create a basic set of templates.

3. Hide the operating system from the user. The user should not be required to become familiar with *any* UNIX operating system commands. File maintenance and system start-up/restart and

backup procedures should all be menu-driven events. It must not be assumed that the operators are computer literate or that they will become familiar with the UNIX operating environment.

4. Incorporate a speaker volume meter. Whether this is accomplished via software or some temporary hardware solution is unimportant. The primary concern is that users learn what speech volume is necessary to train and use the system correctly.

5. Improve procedures for starting status board display application. The current series of commands necessary to start the application needs to be simplified to a single menu selection. Requiring a series of commands to be entered on a variety of terminals is both confusing and beyond the capability of most novice users.

6. Tune the recognizer for the CATCC environment. The engineering parameter file should be adjusted for this particular syntax and environment.

Our long-term recommendations, while not considered critical for the development of the prototype, are nonetheless issues of major concern during full-scale development. They are provided as suggestions for future endeavors.

1. Solicit operator input. Individuals (operators) intimately familiar with the environment should be consulted in the implementation of any of the software component interface.

2. Develop the interface in terms familiar to the operator. Avoid at all costs unfamiliar terms or concepts when presenting information to the operator. Eliminate speech technology terms such as "FORCED RECOGNITION FAILURE" or "OPEN RECOGNITION."

3. Ensure that the software is sailor proof. All software components must protect the user from the unpredictability caused by incorrect or unexpected inputs or abnormal execution.

4. Syntax Recommendations

Because of the relative importance of syntax in connected speech systems, we are making the following recommendations. These are considered both short- and long-term suggestions. In

general, the syntax "operated" correctly, but the following recommendations are offered as a means of improving the application and, if possible, should be incorporated into the prototype syntax:

1. Ensure syntactic correctness. The present syntax does not accurately reflect valid phrases.

2. Allow for error correction. As discussed in Chapter II, a variety of error-correction schemes should be incorporated.

3. Implement task-specific syntaxes. Our research demonstrated that smaller, more specific syntaxes performed significantly better. The application should be designed such that a unique syntax is available for each of the displays.

4. Solicit user input in the syntax design process. A system for verbally communicating status board information presently exists with the manual system. The operators should be involved in developing the syntaxes consistent with their current approach.

C. CONCLUSIONS

Based on our military experience, the extensive "hands on" experience with the prototype system, and the collective opinions of our test subjects, we have developed three significant conclusions.

First, we believe that the input, display, and dissemination of aircraft status information aboard an aircraft carrier is a process which can be more efficiently and effectively accomplished using automation. We are not alone in our opinion; other carriers are already using microcomputers to manage and display CATCC information in a very similar application [Ref. 24]. There is no doubt that the potential exists to dramatically increase the accuracy and timeliness of this critical information throughout the ship.

Second, voice recognition technologies offer an input mechanism which appears well-suited to the CATCC environment. We believe that with training, proper equipment, and well-designed software, a voice-based automated display system could be effectively implemented. Our research demonstrated that even with minimal training, and despite significant software difficulties, we were able to achieve acceptable recognition rates in a noisy environment.

Finally, if the short-term recommendations are adopted, the prototype can, and should, be tested aboard an operational aircraft carrier as a means of validating and demonstrating the concept outside the protective shelter of a laboratory.

APPENDIX A

AIR OPS AND CATCC STATUS BOARDS

TIMEZONE I ±7 ZULU

	WX		ALT
	POSIT		NEARLAND
TRAPS TODAY	DAY	NIGHT	
TRAPS TO DATE	DAY	NIGHT	

EVT ______ CASE DEF ______ CASE REC ______ MAG VAR ______ HELC

DIVERT	RWY	TAC	BRG \ DIST	TIME ______	C/S	

C/S	SQD	A/C	S/N	TK D/N	HDN	IFR	FP D/N	VFR	TKR STAT		

SMALLBOY ______ ID/TAC ______

AIR OPS #1

EVT CASE RAMP MR EMR BRC/FB

EVT	SIDE	PILOT	OFF	ON	SIDE	ST/TM	REMARKS	B

AIR OPS #2 AND #3

ALERT STATUS

A

ALERT	SIDE	PILOT	SET	ALERT	SIDE	PILOT	SET	
								FIRE FLOODING
								X7888
								MEDICAL
								X7911

DIVERT STATUS

SIDE	ATD	ETA	DEST	FPP	SIDE	ATD	ETA	DEST	FPP

B

AIR OPS #4

EQUIPMENT STATUS								
OJ374			RADAR	OD746	NAV STATUS	RZ + 1S	19 mc	RD 379
CH	TX	RX	SPN 41 ____	APP A ____	NTDS ____	14 ____ 14 ____	21 mc	REC ____
1			SPN 42 A ____ B ____	APP B ____	KCS ____	JX	22 mc	PLAY ____
2			SPN 43 ____	DEP ____	SINS ____	2JC	42 mc	TIME ____
3			SPS 49 ____	MAR ____	MK 19 ____	11JC	TACAN	PLAT
4			SPS 10 ____	SUP ____	FWD ____	DAIR	1 ____	CAMERA
5			SPA 18 A ____ B ____		AFT ____	A ____ B ____	2 ____	NDB
6			EMCON		MOD/LINES			
7								
8			HERC		MOD/LINES			
9								
10			15G21					

RR	PHONE	SQDN	REMARKS
1	7401		
2	7402		
3	7403		
4	7404		
5	7405		
6	7406		
7	7407		
8	7408		
9	7409		

b

(SAME AS AIR WAR NO. 4a)

AIR OPS #5

GENERAL INFORMATION BOARD

	Bingo Fuel				Wx		Time	
	SQDN	A/C	HD	IFR	Alt			B
					IB	BRG		
					FB			
A					Case Recovery			
					CV	Approach		
					Primary Divert			
					Time	BRG	DIST	
					Wx			C
					Secondary Divert			
					Time	BRG	DIST	
	Distance		NM		Wx			

SQDN	A/C	L4	SQDN	A/C	L4	SQDN	A/C	L4	
									D

CCA BOARD #1

. DELTA BOARD

TYPE A/C	LOW DELTA	HIGH DELTA	HD	SQDRN	A/C	IFR	VFR

CCA STATUS BOARD #2

MARSHAL OVERFLOW BOARD

OB | RAMP | BRC | IH

	SIDE	ST/TM	RAD	DME	ANG	EAT	BN	HD/TM	COM/TM	M	SIDE	REMARK
WX												
ALT												
CASE												
CV												
FB												

LONGEST NAVAID

ID	TAC	RWY	STATUS	FREQ

SMALLBOY	ID	TAC

ASW DATUM

TYPE	C/S	BRG/DIST	REM

CCA CHAN	BVT	FREQ	CCA CHAN	BVT	FREQ	TACAN
1			6			NDB
2			7			BULLSEYE
3			8			ACLS
4			9			CHAN A ____ FOR MODE ____ APPR.
5			10			CHAN B ____ FOR MODE ____ APPR.

CCA BOARD #3

MARSHAL STATUS BOARD

CASE CV FB ALT

SIDE	ST/TM	RAD	DME	ANG	EAT	BN	HD/TM	COM/TM	M	SIDE	RMK

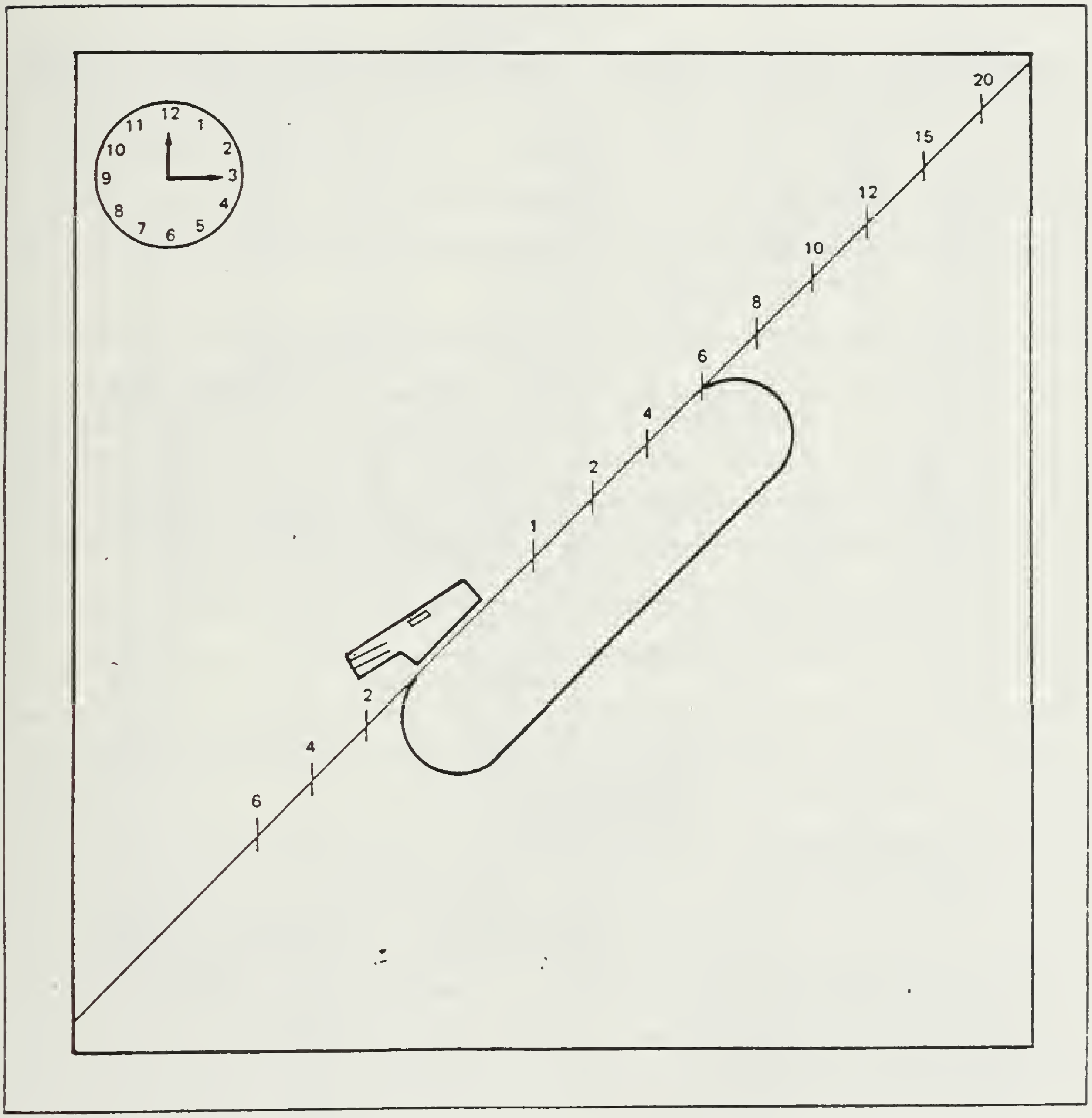
CCA BOARD #4

APPROACH STATUS BOARD

FB DW

SIDE	BN	10	D	FRC	L/ON	MODE	ON	ST/TM	SIDE	REMARK

CCA BOARD #5 AND #6



CCA NO. 7 ("GLOW" BOARD)

DIVERT BOARD

EVENT

SIDE	√IN	A/B	REMARKS

TANKERS

SIDE	ST/TM	GIVE	ANG	BN	RMKS

HELOS SIDE	OFF	SPLASH TM	BN	RMKS

CASE: LAUNCH: ACT:

BRC: WX:

REF RAD:

PRIMARY DIVERT

BEARING	DISTANCE	TIME

WX:

SECONDARY DIVERT

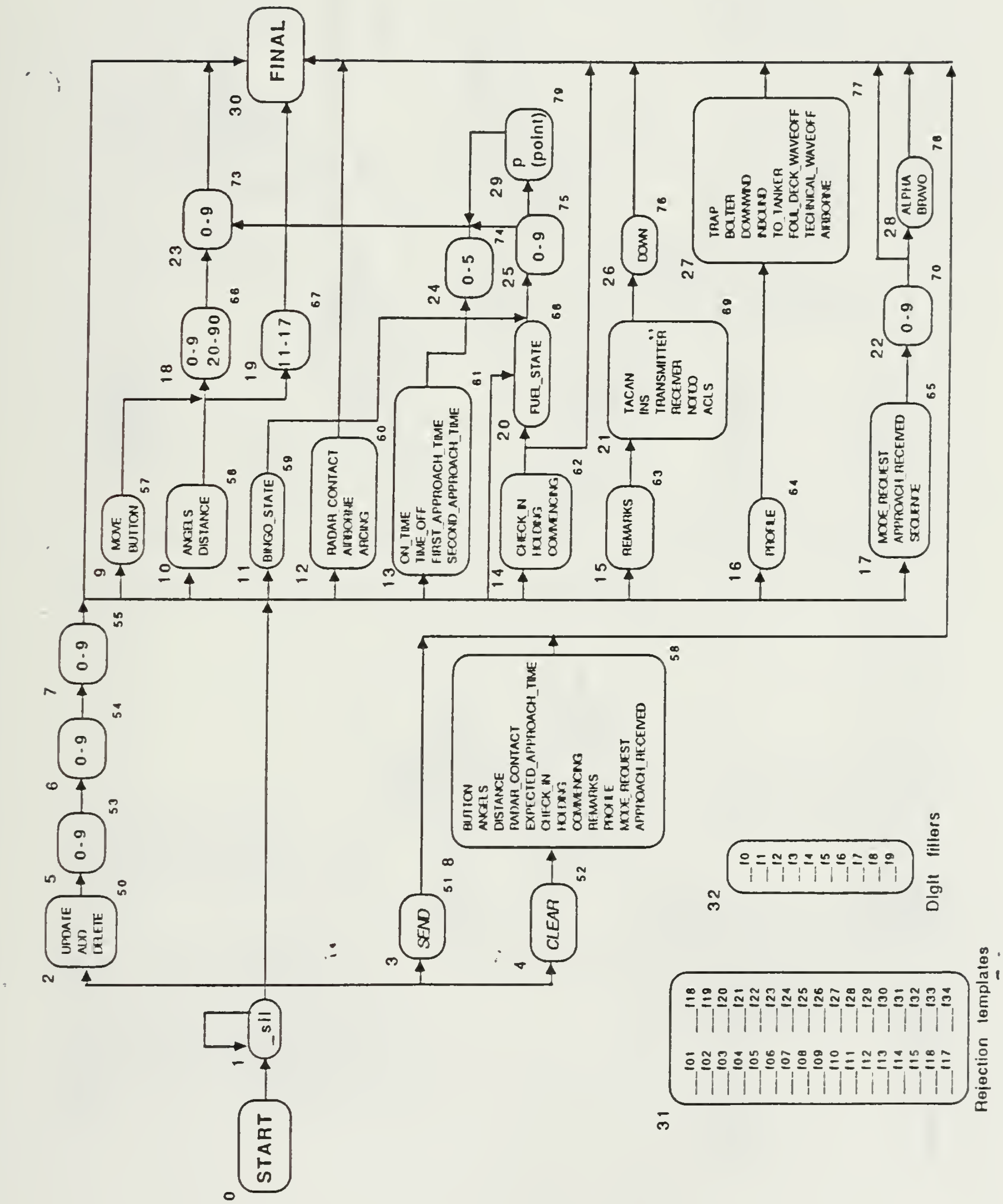
BEARING	DISTANCE	TIME

WX:

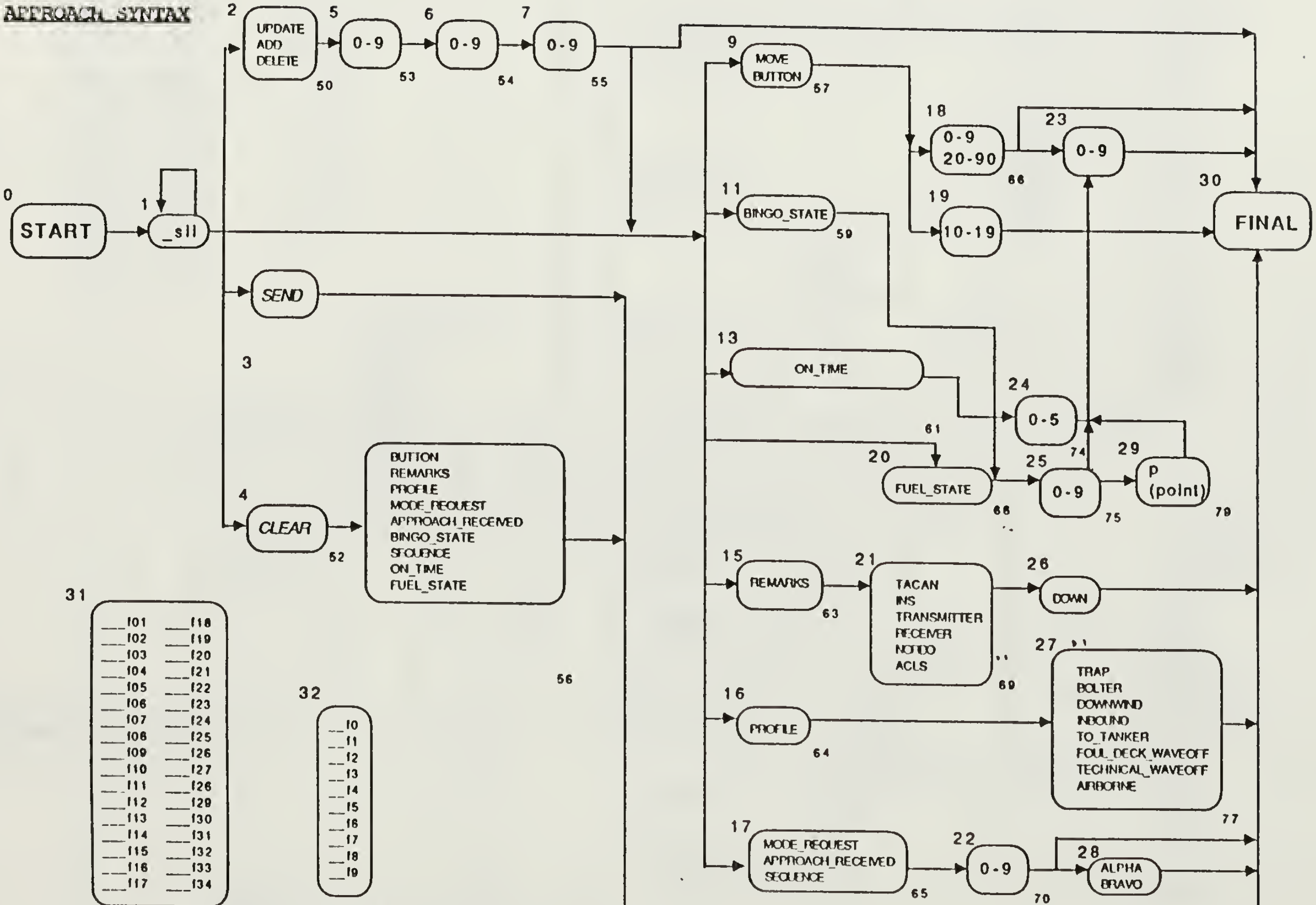
CCA BOARD #8

APPENDIX B

CATCC SYNTAXES



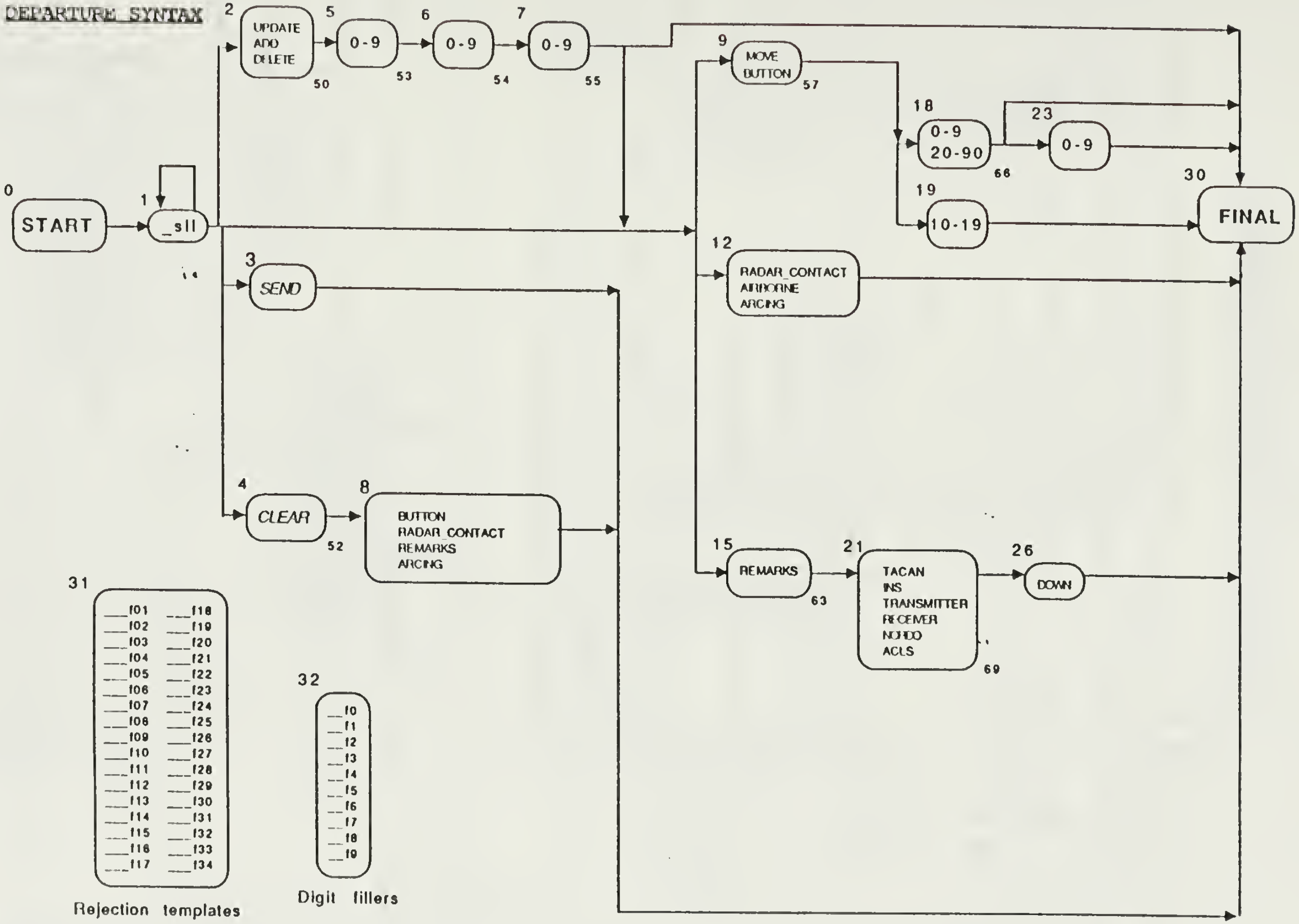
APPROACH_SYNTAX



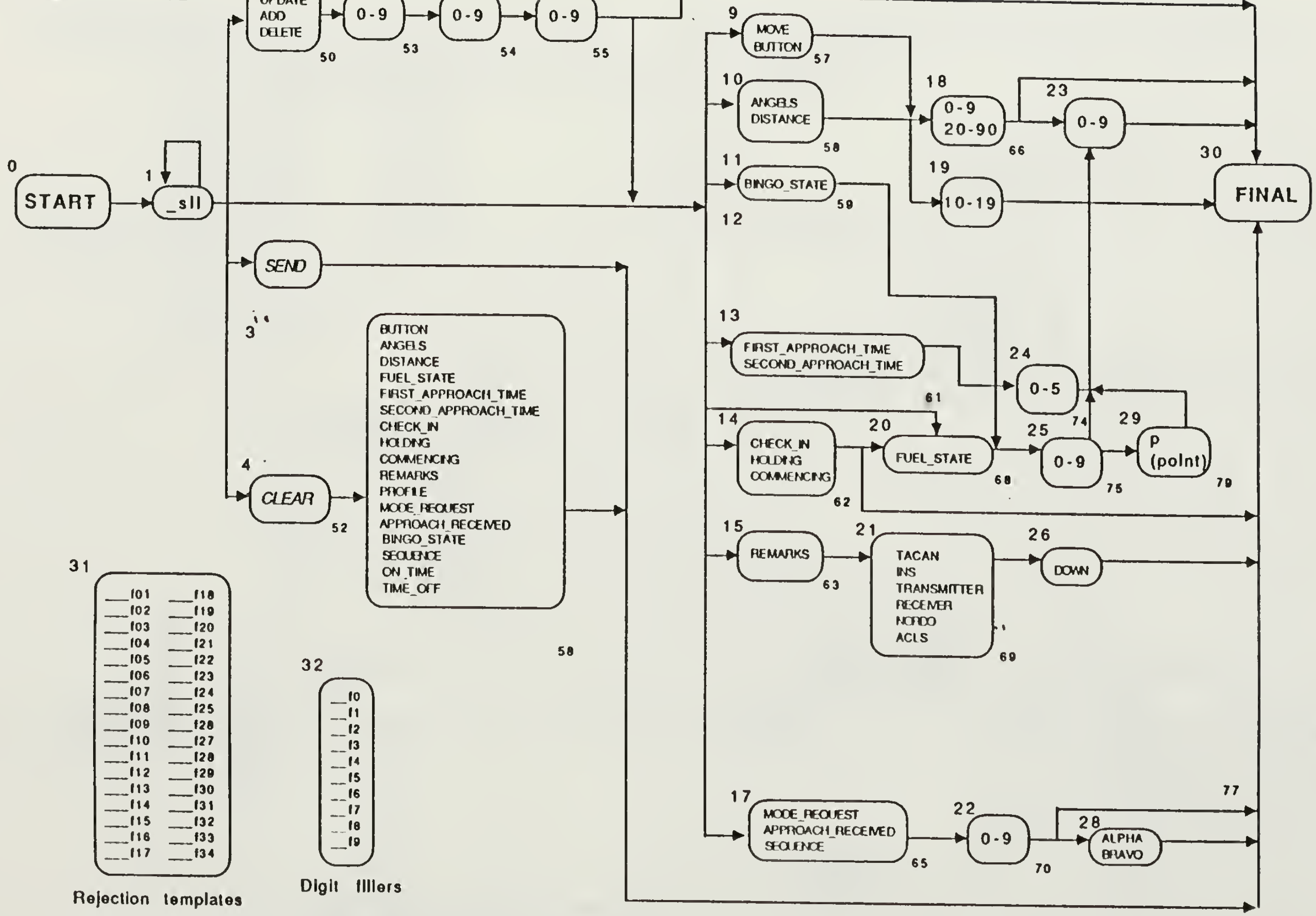
Rejection templates

Digit fillers

DEPARTURE SYNTAX



MARSHAL_SYNTAX



APPENDIX C

MASTER INSTRUCTION SHEET

1. To conduct a test, first set up, at a minimum, the first four tests. To do this, open a window, type the phrase, and then, using the sun-tools pull-down menu, "close" the window. All eight of the following hostpump commands will be used. The only change is to substitute the user's initials for "INIT":

```
hostpump  INIT.asyn.q approach.pump approach.syn
hostpump  INIT.dsyn.q departure.pump departure.syn
hostpump  INIT.msyn.q marshall.pump marshall.syn
hostpump  INIT.csyn.q combined.pump combined.syn

hostpump  INIT.asyn.n approach.pump approach.syn
hostpump  INIT.dsyn.n departure.pump departure.syn
hostpump  INIT.msyn.n marshall.pump marshall.syn
hostpump  INIT.csyn.n combined.pump combined.syn
```

2. Now train the user as usual (using "host"). At the completion of the training and prior to running any recognition, you MUST go to sd0/newtrain/NOSC6/TEMPLATE and execute the following:

```
cp subject_last_name/point.subject_initials
subject_last_name/p.subj_init
```

EX:

```
cp spegele/point.js spegele/p.js
```

NOTE: If you don't do this, you will get an error message when loading hostpump.

3. Now exit host and set up to conduct the test. If noise is required, activate the window with "input trainer" and set for 1 sec. Also, don't forget to turn on the radio beside you. Take a noise level reading and record the test results.

4. Using a copy of the file, annotate who the subject is, any training difficulties (problem words, etc.), time of day, and any substitution or misspeak errors.

NOTES:

Turn Dectalk off during quiet tests

Dectalk setting 5 o'clock for 75 dBA

Subject brief:

mic positioning
speaking rate (speed)
speaking style (normal)
give example

APPENDIX D

TEST SUBJECT INFORMATION SHEET

NAME: ____________________________

TIME START: ____________________

TRAINING COMPLETE: ___________

TEST START: ____________________

TEST COMPLETE: _______________

PROBLEMS:

APPENDIX E

TRAINING VERIFICATION SHEET

UPDATE 0 0 0
UPDATE 1 1 1
UPDATE 2 2 2
ADD 3 3 3
ADD 4 4 4
ADD 5 5 5
DELETE 6 6 6
DELETE 7 7 7
DELETE 8 8 8
DELETE 9 9 9

CLEAR BUTTON
CLEAR ANGELS
CLEAR DISTANCE
CLEAR RADAR CONTACT
CLEAR FIRST APPROACH TIME
CLEAR SECOND APPROACH TIME
CLEAR CHECK IN
CLEAR HOLDING
CLEAR COMMENCING
CLEAR REMARKS
CLEAR PROFILE
CLEAR MODE REQUEST
CLEAR APPROACH RECEIVED
CLEAR BINGO STATE
CLEAR SEQUENCE
CLEAR AIRBORNE
CLEAR ARCING
CLEAR ON TIME
CLEAR TIME OFF

CHECK IN FUEL STATE 5 P 9
CHECK IN FUEL STATE 1 P 4
CHECK IN FUEL STATE 3 P 8

PROFILE TRAP
PROFILE BOLTER
PROFILE DOWNWIND
PROFILE INBOUND

PROFILE TO TANKER
PROFILE FOUL DECK WAVEOFF
PROFILE TECHNICAL WAVEOFF
PROFILE AIRBORNE

MODE REQUEST 5 ALPHA
MODE REQUEST 3 ALPHA
MODE REQUEST 8 ALPHA

APPROACH RECEIVED 2 BRAVO
APPROACH RECEIVED 0 BRAVO
APPROACH RECEIVED 1 BRAVO

SEQUENCE 4 ALPHA
SEQUENCE 9 ALPHA
SEQUENCE 6 ALPHA

REMARKS TACAN DOWN
REMARKS INS DOWN
REMARKS TRANSMITTER DOWN
REMARKS RECEIVER DOWN
REMARKS NORDO DOWN
REMARKS ACLS DOWN

APPENDIX F

TESTING MATRIX

SUBJECT	CONDITION separate/ noise	combined/ noise	combined/ quiet	separate/ quiet
SUBJECT 2	1	2	3	4
SUBJECT 4	4	1	2	3
SUBJECT 8	3	4	1	2
SUBJECT 10	2	3	4	1
SUBJECT 3	1	2	3	4
SUBJECT 5	4	1	2	3
SUBJECT 11	1	4	3	2
SUBJECT 7	4	3	2	1
SUBJECT 6	3	2	1	4
SUBJECT 9	2	1	4	3
SUBJECT 12	2	3	4	1
SUBJECT 1	1	4	3	2

APPENDIX G

SUBJECT INSTRUCTIONS

1. There are four files which we will, during the conduct of the test, ask you to call up. In order to display the contests of a file, you must type "more filename.extension" where the filename and extension are one of the below listed:

approach.pump

departure.pump

marshall.pump

combined.pump

During the test, you may have to scroll through the file to display phrases not initially shown. To do this, first turn off the microphone, then hit the carriage return until you see "E N D O F T E S T." Then turn the mike on. To leave the file, continue depressing the carriage return until you are returned to the UNIX prompt "tamale=/usr.MC68020/sd0/stat/SCENARIO."

2. Phrases read from the test file should be read in the same manner as you practiced; a short 1-3 sec. pause is sufficient between phrases. There is no need to rush the reading and you should not be concerned with exceeding the speed of the voice recognizer.

3. You may leave the microphone open [ON] during all phases of training and testing. If you feel a need to momentarily pause, then you

should turn the microphone off until ready to resume voice recognition.

4. If during the test you inadvertently misspeak and realize your error, then:

- Turn the microphone off;
- Alert the tester;
- Turn the microphone on;
- Repeat the phrase (correctly);
- Continue the test.

APPENDIX H

TEST FILES

COMBINED SYNTAX

ADD 5 7 5 PROFILE TRAP

DELETE 9 1 4 PROFILE BOLTER

CLEAR PROFILE

BINGO STATE 8 POINT 8

CLEAR SEQUENCE

PROFILE DOWNWIND

FUEL STATE 6 POINT 7

CLEAR APPROACH RECEIVED

UPDATE 0 6 0 PROFILE TO TANKER

PROFILE INBOUND

DELETE 0 8 1 BINGO STATE 7 POINT 6

ON TIME 5 5

DELETE 7 1 5

DELETE 3 3 3 FUEL STATE 1 POINT 9

CLEAR FUEL STATE

ADD 2 4 9 FUEL STATE 2 POINT 0

CLEAR ON TIME

FUEL STATE 5 POINT 2

CLEAR MODE REQUEST

UPDATE 5 5 4 PROFILE FOUL DECK WAVEOFF

ADD 4 8 2 PROFILE TECHNICAL WAVEOFF

ON TIME 5 4

CLEAR BINGO STATE

UPDATE 3 2 1 ON TIME 2 3

ADD 6 5 2 AIRBORNE

REMARKS TRANSMITTER DOWN

ADD 3 4 1 RADAR CONTACT

CLEAR ARCING

UPDATE 8 0 8 TIME OFF 4 2

CLEAR BUTTON

DELETE 7 8 9 REMARKS INS DOWN

MOVE 16

UPDATE 1 3 3 MOVE 5

DELETE 2 2 7 ARCING

REMARKS NORDO DOWN

CLEAR REMARKS

ADD 4 9 6 REMARKS ACLS DOWN

TIME OFF 3 9

REMARKS RECEIVER DOWN

MOVE 1 5

SEND

CLEAR RADAR CONTACT

UPDATE 9 1 8 BUTTON 11

ADD 8 7 6 ARCING

REMARKS TACAN DOWN

AIRBORNE

UPDATE 2 3 7 MOVE 20 7

BUTTON 19

CLEAR DISTANCE

UPDATE 4 7 8 ANGELS 3

CLEAR SECOND APPROACH TIME

DELETE 1 0 7 CHECK IN FUEL STATE 4 POINT 1

CLEAR HOLDING

ADD 6 5 6 FIRST APPROACH TIME 3 0

CLEAR COMMENCING

UPDATE 5 6 0 APPROACH RECEIVED 9 ALPHA

SECOND APPROACH TIME 4 5

UPDATE 8 2 2 REMARKS TACAN DOWN

ADD 9 9 4 COMMENCING FUEL STATE 3 POINT 6

CLEAR FIRST APPROACH TIME

HOLDING FUEL STATE 9 POINT 5

DISTANCE 2 8

CLEAR ANGELS

ADD 7 0 0 HOLDING

CLEAR CHECK IN

ADD 6 6 9 HOLDING FUEL STATE 4 POINT 7

DELETE 0 0 0 SEQUENCE 0 BRAVO

DELETE 1 9 3 MODE REQUEST 7 BRAVO

BINGO STATE 3 POINT 2

UPDATE 0 4 5 SEQUENCE 5 ALPHA

DISTANCE 30 6

CLEAR TIME OFF

APPROACH SYNTAX

ADD 5 7 5 PROFILE TRAP

DELETE 9 1 4 PROFILE BOLTER

CLEAR PROFILE

BINGO STATE 8 POINT 8

CLEAR SEQUENCE

PROFILE DOWNWIND

FUEL STATE 6 POINT 7

CLEAR APPROACH RECEIVED

UPDATE 0 6 0 PROFILE TO TANKER

PROFILE INBOUND

DELETE 0 8 1 BINGO STATE 7 POINT 6

ON TIME 5 5

DELETE 7 1 5

DELETE 3 3 3 FUEL STATE 1 POINT 9

CLEAR FUEL STATE

ADD 2 4 9 FUEL STATE 2 POINT 0

CLEAR ON TIME

FUEL STATE 5 POINT 2

CLEAR MODE REQUEST

UPDATE 5 5 4 PROFILE FOUL DECK WAVEOFF

ADD 4 8 2 PROFILE TECHNICAL WAVEOFF

ON TIME 5 4

CLEAR BINGO STATE

UPDATE 3 2 1 ON TIME 2 3

DEPARTURE SYNTAX

ADD 6 5 2 AIRBORNE

REMARKS TRANSMITTER DOWN

ADD 3 4 1 RADAR CONTACT

CLEAR ARCING

UPDATE 8 0 8 TIME OFF 4 2

CLEAR BUTTON

DELETE 7 8 9 REMARKS INS DOWN

MOVE 16

UPDATE 1 3 3 MOVE 5

DELETE 2 2 7 ARCING

REMARKS NORDO DOWN

CLEAR REMARKS

ADD 4 9 6 REMARKS ACLS DOWN

TIME OFF 3 9

REMARKS RECEIVER DOWN

MOVE 1 5

SEND

CLEAR RADAR CONTACT

UPDATE 9 1 8 BUTTON 11

ADD 8 7 6 ARCING

REMARKS TACAN DOWN

AIRBORNE

UPDATE 2 3 7 MOVE 20 7

BUTTON 19

MARSHAL SYNTAX

CLEAR DISTANCE

UPDATE 4 7 8 ANGELS 3

CLEAR SECOND APPROACH TIME

DELETE 1 0 7 CHECK IN FUEL STATE 4 POINT 1

CLEAR HOLDING

ADD 6 5 6 FIRST APPROACH TIME 3 0

CLEAR COMMENCING

UPDATE 5 6 0 APPROACH RECEIVED 9 ALPHA

SECOND APPROACH TIME 4 5

UPDATE 8 2 2 REMARKS TACAN DOWN

ADD 9 9 4 COMMENCING FUEL STATE 3 POINT 6

CLEAR FIRST APPROACH TIME

HOLDING FUEL STATE 9 POINT 5

DISTANCE 2 8

CLEAR ANGELS

ADD 7 0 0 HOLDING

CLEAR CHECK IN

ADD 6 6 9 HOLDING FUEL STATE 4 POINT 7

DELETE 0 0 0 SEQUENCE 0 BRAVO

DELETE 1 9 3 MODE REQUEST 7 BRAVO

BINGO STATE 3 POINT 2

UPDATE 0 4 5 SEQUENCE 5 ALPHA

DISTANCE 30 6

CLEAR TIME OFF

APPENDIX I

CATCC RADIO CALLS

1. Tarhat Marshal, this is Redstone one zero two in company with one zero three on your two four five radial at forty six miles, angels twenty seven, low state eight point three, over.

2. Redstone one zero two, marshal. This will be a case three recovery, altimeter two niner niner two. Redstone one zero two, marshal two five zero for twenty three, angels eight. Expect approach time four eight, approach button one six, time now two four and one quarter, over.

3. Redstone one zero two, roger.

4. Redstone one zero three, marshal and two five zero for twenty four, angels niner, expect approach time four niner, approach button one eight, time now two four and one half, over.

5. Redstone one zero three, roger.

6. Marshal, this is two one three with two one four in company, on your three zero five for thirty three, angels twenty three, low state eight point six, requesting mode two's.

7. City Desk two one three, marshal, case three recovery, altimeter two niner niner two, marshal two five zero radial, at twenty five, angels ten, expect approach button one six, time now two seven and one quarter, over.

8. City Desk two one three, roger.

9. City Desk two one four, marshal two five zero radial at twenty six, angels eleven, expect approach time five one, approach button one eight, time now three zero and one half, over.

10. City Desk two one four, roger.

11. Marshal, Canasta four zero zero checking in with play mate four zero four on your two zero zero radial at thirty one, angels twenty six, low state six point two, requesting mode one alpha's.

12. Canasta four zero zero, Marshal, case three recovery, altimeter two niner niner two. Canasta four one zero marshall two five zero, twenty one, angels six, expect approach time four six, approach button one six, time now three one.

13. Canasta four zero zero, roger.

14. Canasta four zero four, marshal two five zero for twenty two, angels seven, expect approach time four seven, approach button one eight, time now three one and one half.

15. Canasta four zero four roger, button one eight.

20. Ten seconds,
Five
Four
Three
Two
One
Mark, time three three.

21. Marshal, Redstone one zero two in holding, angels eight, state seven point nine.

22. Redstone one zero two, roger, angels eight.

23. Marshal, Redstone one zero three in holding, angels niner, state eight point zero.

24. Redstone one zero three, roger, angels niner.

28. Marshal, Canasta four zero zero in holding, angels six, state four point one.

29. Canasta four zero zero, roger.

35. Marshal, City Desk two one three in holding angels ten, state eight point one.

36. City Desk two one three, roger say mode requested.

37. Mode two.

43. Marshal, City Desk two one four, established, angels eleven, state eight point zero, request mode two.

44. City Desk two one four, roger.

48. Canasta four zero four established, angels seven, state four point five.

49. Canasta four zero four, roger.

53. Ten seconds until time four three.

54. Five
Four
Three
Two
One
Mark, time four three.

59. Marshal, Canasta four zero zero commencing, state three point four.

60. Canasta four zero zero, radar contact twenty one miles, final bearing zero seven zero.

62. Canasta four zero zero, platform.

63. Canasta four zero zero, go button one ix.

66. Canasta four zero four commencing, state three point three.

67. Canasta four zero four, radar contact twenty two miles, final bearing zero seven zero.

68. Canasta four zero four, platform.

69. Canasta four zero four, go button on eight.

70. Redstone one zero two commencing, state six point four.

71. Redstone one zero two radar contact twenty three miles, final bearing zero seven zero.

72. Ninety nine Tarhat, altimeter two niner niner five.

73. Redstone one zero three commencing, state six point zero.

74. Redstone one zero three, radar contact twenty four mils, final bearing zero seven zero.

75. Redstone one zero two, platform.

76. Roger.

77. Redstone one zero two, go button one six.

78. Redstone one zero two, switching.

80. Magic six zero four, roger.

81. Marshal, City Desk two one three commencing, state five point six.

82. City Desk two one three, radar contact twenty five miles, final bearing zero seven zero.

83. Redstone one zero three, platform.

84. Redstone one zero three, roger.

85. Redstone one zero three, go button one eight.

86. One zero three, switching.

87. City Desk two one three, platform.

88. Roger.

89. Marshal, City Desk two one four commencing, state five point five.

90. City Desk two one four, radar contact twenty six miles, final bearing zero seven zero.

91. City Desk two one three, go button one six.

APPENDIX J

RESPONSE PHRASE SAMPLE FILE

NOTE: Taken from Subject 1 Quiet (0 dBA) and Separate (Marshal) condition.

WORD	WORD SCORE	REJECTION SCORE
CLEAR	24	18
DISTANCE	20	18
UPDATE	15	25
4	18	23
7	35	28
8	12	25
ANGELS	23	47
30	23	21
CLEAR	28	32
SECOND_APPROACH_TIM	23	23
DELETE	15	32
1	21	32
0	39	44
7	22	23
CHECK_IN	19	28
FUEL_STATE	19	33
4	19	14
*P	60	37
1	20	14
*CLEAR	37	16
HOLDING	22	33
ADD	13	41
6	13	16
5	22	19
6	14	17
FIRST_APPROACH_TIME	19	28
3	25	21
0	37	37

CLEAR	16	21
COMMENCING	16	20
UPDATE	12	23
5	30	32
6	11	11
0	49	45
APPROACH_RECEIVED	22	26
5	42	37
ALPHA	19	32
SECOND_APPROACH_TIM	17	23
4	23	21
5	21	20
UPDATE	13	19
8	11	25
2	19	21
2	21	31
REMARKS	27	23
TACAN	13	39
DOWN	16	27
ADD	11	43
9	28	25
9	33	2
4	26	23
COMMENCING	23	24
FUEL_STATE	16	26
3	21	20
p	24	24
6	14	18
CLEAR	23	32
FIRST_APPROACH_TIME	15	23
HOLDING	26	36
FUEL_STATE	20	29
9	36	31
p	26	23
5	22	30
DISTANCE	24	26
2	21	24
9	27	27

CLEAR	30	24
ANGELS	31	41
ADD	11	40
7	20	20
0	37	42
0	47	37
HOLDING	21	27
CLEAR	22	27
CHECK_IN	17	27
COMMENCING	13	12
ADD	9	43
6	13	15
6	13	14
	34	30
HOLDING	22	37
FUEL_STATE	17	29
4	16	13
*p	40	22
7	22	28
DELETE	12	32
0	34	41
0	39	47
0	43	44
SEQUENCE	25	22
0	37	26
BRAVO	17	25
DELETE	17	34
1	23	25
9	49	47
3	30	28
MODE_REQUEST	20	28
7	24	29
BRAVO	18	27
BINGO_STATE	13	24
3	20	23
p	23	21
2	19	32
UPDATE	13	23
*0	35	18
4	24	13
5	19	21

SEQUENCE	26	23
5	31	25
ALPHA	20	27
DISTANCE	18	22
30	18	21
6	16	17
CLEAR	23	24
TIME_OFF	13	32
SEQUENCE	14	13
SEQUENCE	12	12

APPENDIX K

CATCC VOICE RECOGNITION POST-TEST QUESTIONNAIRE

1. What is your curriculum? #_____ Descriptor __________
2. To which service do you belong? __________ (i.e., USN, etc.)
3. What is your grade? __________ (i.e., O-2, O-5, etc.)
4. Are you a Naval Aviator? Yes _____ No _____
5. Are you a Naval Flight Officer? Yes _____ No _____
6. Have you any previous experience with voice recognition systems? If yes, how may hours (approx.)? ______. Mark "0" if no experience.
7. Based on your previous training and work experience, how comfortable or uncomfortable were you with the vocabulary used in this experiment?

   ____ Very comfortable

   ____ Comfortable

   ____ Borderline

   ____ Uncomfortable

   ____ Very uncomfortable
8. Based on your previous training and work experience, how comfortable or uncomfortable are you utilizing a microphone?

   ____ Very comfortable

   ____ Comfortable

   ____ Borderline

____ Uncomfortable

____ Very uncomfortable

9. Have you ever been assigned to a CATCC or CIC? Yes ___ No ___

   If yes, how many months? _____ (mos.)

   If no, have you ever been exposed to CATCC/CIC operations?

   Yes ______ No ______

   If yes, now long? _____ hours, weeks, months (circle one).

10. The training session, as guided by the experimenter, was:

    ____ Very easy

    ____ Quite easy

    ____ Fairly easy

    ____ Borderline

    ____ Fairly difficult

    ____ Quite difficult

    ____ Very difficult

11. The quality of the Sun workstation display used for training was:

    ____ Excellent

    ____ Good

    ____ Only fair

    ____ Poor

    ____ Terrible

12. The quality of the WYSE display used for testing was:

    ____ Excellent

    ____ Good

    ____ Only fair

____ Poor

____ Terrible

13. How satisfied were you with the ergonomics of the microphone set?

____ Very satisfied

____ Satisfied

____ Borderline

____ Dissatisfied

____ Very dissatisfied

14. How acceptable or unacceptable do you feel voice input technology is for the CATCC or CIC environment?

____ Completely acceptable

____ Reasonably acceptable

____ Borderline

____ Moderately unacceptable

____ Extremely unacceptable

15. If you were responsible for the operation of a CATCC or CIC, how would you accept a fully developed voice input status board system to replace the current methodology?

____ Without hesitation

____ With little hesitation

____ With some hesitation

____ With great hesitation

16. What do you feel are the major issues (pro and/or con) with regard to utilizing voice input in the CATCC/CIC?

17. What other areas, if any, in the Armed Services do you see where voice input could be used?

APPENDIX L

OPEN-ENDED QUESTION RESULTS FROM POST-TEST QUESTIONNAIRE

What do you feel are the major issues (pro and/or con) with regard to utilizing voice input in the CATCC/CIC?

Reliability. Keeping the thing up.

Quality of displays.

Noise susceptibility.

Training and turnover of various personnel to system.

Rapid replacement of personnel at a station during battle, "Killed—now replace in midst of battle situation."

Training of users—microphone fear.

Control of environmental noises that are quite prevalent.

Training

System maintenance.

Operation in degraded or unusual conditions.

Ability to revert to manual system over long term (lost skills).

Noise level is much higher in a carrier than it was in the booth.

Standardizing key words and phrases may be difficult.

Making system reliable (error rate low).

Making system sailor-proof (rugged).

Educating Navy to benefits.

Reliability.

Pro—more readable and faster update of information on (status) boards, possible space savings.

Faster, more accurate data.

Stress.

Background noise interference.

Overlapping duty sections (changing over of personnel).

Fatigue.

Pro—Free person from writing status on board; faster than writing.

Con—Interpreted incorrectly; able to respond in varying noise environments.

Reliability.

Ease of training.

Effect of flight-op noise.

Back-up when it fails.

Distinction in voices due to colds.

What other areas, if any, in the Armed Services do you see where voice input could be used?

Cockpits of all types of A/C (aircraft).

Rapid strike coordination messages, surface to subsurface.

ASWMOD (coordination of antisubmarine warfare assets).

CIC (Combat Information Center)

Aircraft.

NTDS (Navy Tactical Display Systems)

Onboard aircraft (routine duties).

Input to flight navigation systems.

Message preparation.

Briefs, presentations.

Other types of status board maintenance.

Command and control for unmanned vehicles.

Testing.

Quick display information updates.

Anywhere status updates, etc. are manually recorded and consist of a finite set of words.

Software development—input can be much faster with voice recognition than by keyboard.

Security checkpoints (possibly).

Aircraft—to ease button smashing mode.

HUD (Heads Up Display) interface for coming aboard the ship, e.g., "SAY ALTITUDE" without leaving the meatball (the marker for landing successfully aboard the aircraft carrier).

LIST OF REFERENCES

1. Taggart, J. L., and Wolfe, C. D., *Voice Recognition as an Input Modality for the Tacco Preflight Data Insertion Task in the P-3C Aircraft*, Masters Thesis, Naval Postgraduate School, Monterey, CA, March 1981.

2. Lea, W. A., *Trends in Speech Recognition*, Prentice-Hall, 1980.

3. Reddy, D. R., "Speech Recognition by Machine: A Review," *Proceedings of the IEEE*, v. 64, April 1976.

4. Woodard, J. P., and Cupples, E. J., "Selected Military Applications of Automatic Speech Recognition Technology," *IEEE Communications Magazine*, December 1983.

5. Martin, T. B., "Practical Application of Voice Input to Machines," *Proceedings of the IEEE*, v. 64, April 1976.

6. Beek, B., and others, "An Assessment of the Technology of Automatic Speech Recognition for Military Applications," *IEEE Transactions on Acoustics, Speech and Signal Processing*, v. ASSP-25, August 1977.

7. North Carolina State University under Contract Number DAAC 29-81-D-0100, *An Evaluation of Speech Recognition Technology*, by M. G. Joost, T. S. Moody, and R. D. Rodman, 5 December 1986.

8. Werkowitz, E., "Speech Recognition in the Tactical Environment: AFTI 16 Voice Command Flight Test," in *Proceedings of the Conference "Speech Tech '84,"* 1984.

9. Grishman, R., *Computational Linguistics*, Cambridge University Press, 1986.

10. Obermeier, K. K., "Natural-Language Processing," *Byte*, v. 12, no. 14, December 1987.

11. Winograd, T., *Language as a Cognitive Process, Vol. 1: Syntax*, Addison-Wesley, 1983.

12. Allen, J., *Natural Language Understanding*, Benjamin/Cummings, Inc., 1987.

13. Kaplan, R. M., "Augmented Transition Networks as Psychological Models of Sentence Comprehension," *Artificial Intelligence*, v. 3, 1972.

14. Miller, G., "Magical Number Seven, Plus or Minus Two," *The Psychological Review*, vol. 3, no. 2, March 1956.

15. Plutchak, B., *User Interface for Voice Actuated Status Boards for the CATCC*, paper presented at Speech Technology Symposium, New York City, 23 April 1988.

16. *ITTDCD VRS 1280 Product Description*, ITT Defense Communications Division, 6 November 1987.

17. *ITT-DCD's Model 1280VRS User's Manual*, ITT Defense Communications Division, 19 May 1987.

18. Rodman, R. D., Joost, M. G., and Moody, T. S., "Performance Evaluation of Connected Speech Recognition Systems," *Proceedings Speech Tech '87*, 1987.

19. Pallett, D. S., "Performance Assessment of Automatic Speech Recognizers," *Journal of Research of the National Bureau of Standards*, v. 90, no. 5, 1985.

20. Cox, D. R., *The Analysis of Binary Data*, Chapman and Hill, London, 1970.

21. Pressman, R. S., *Software Engineering: A Practitioner's Approach*, McGraw-Hill, Inc., 1987.

22. Poock, G. .K., and Martin, B. J., *An Examination of Some Error Correcting Techniques for Continuous Speech Recognition Technology*, Masters Thesis, Naval Postgraduate School, Monterey, CA, June 1985.

23. Monk, A., ed., *Fundamentals of Human-Computer Interaction*, Academic Press, 1985.

24. Management Information and Instructional Systems Activity, Pensacola, FL, *Aircraft Monitoring Program (AMP)*, user's manual, 31 March 1986.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, CA 93943-5002	2
3.	Director, Information Systems (OP-945) Office of the Chief of Naval Operations Navy Department Washington, DC 20350-2000	1
4.	Superintendent, Naval Postgraduate School Computer Technology Programs, Code 37 Monterey, CA 93943-5000	1
5.	Commandant of the Marine Corps Code TE 06 Headquarters, U. S. Marine Corps Washington, DC 20360-0001	1
6.	Naval Ocean Systems Center Attn: Bruce Plutchak 271 Catalina Blvd. Code 441 San Diego, CA 92152	10
7.	Naval Ocean Systems Center Attn: Dr. Willis Rasmussen 271 Catalina Blvd. Code 44 San Diego, CA 92152	1
8.	Naval Ocean Systems Center Attn: Dr. Eugene Cooper 271 Catalina Blvd. Code 013 San Diego, CA 92152	1

9. Naval Ocean Systems Center 1
Attn: Library
271 Catalina Blvd.
Code 964
San Diego, CA 92152

10. ITT Defense Communications 1
Attn: Howard Koble
10060 Carroll Canyon Road
San Diego, CA 92131

11. LOGICON 1
Attn: Jim Miller
4010 Sorrento Valley Blvd.
P. O. Box 85158
San Diego, CA 92138

12. Commander 1
Naval Air Systems Command
Naval Air Systems Command Headquarters
Attn: Mike Mancini
Code APC205-13
Washington, DC 20361-0001

13. Commander 1
Naval Training Center
Attn: Art Soft
Orlando, FL 32813

14. Dr. Gary K. Poock, Code 55Pk 5
Naval Postgraduate School
Monterey, CA 93943-5000

15. Dr. Vincent Lum, Code 52Lu 1
Naval Postgraduate School
Monterey, CA 93943-5000

16. CAPT Robert D. Jensen 1
SMC 2999
Naval Postgraduate School
Monterey, CA 93943

17. CAPT John J. Spegele 1
Computer Science Department
U. S. Naval Academy
Annapolis, MD 21402